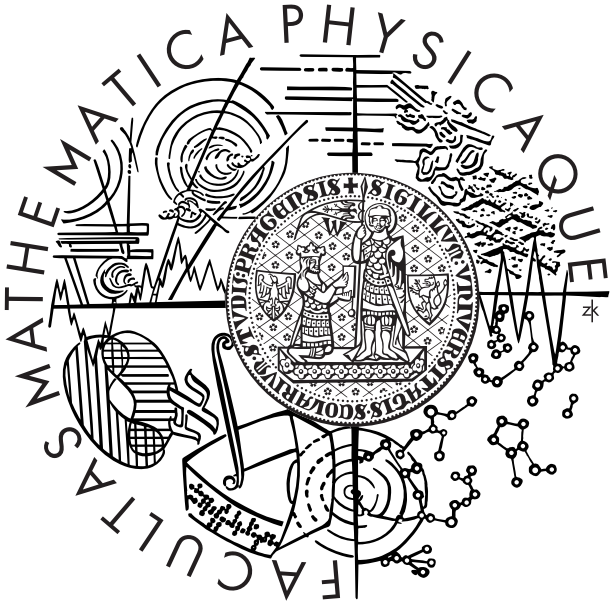
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Tomáš KAZMAR

Opacity Quantification in Cardiac Angiogram Sequence

Katedra softwarového inženýrství
Vedoucí diplomové práce: *Dr. Ing. Jan* KYBIC
Studijní program:
Informatika, Softwarové systémy — Počítačová grafika

PODĚKOVÁNÍ

Děkuji za vedení této práce Dr. Ing. Janu Kybicovi z Centra strojového vnímání, katedry kybernitiky, FEL ČVUT. Za poskytnutí vstupních dat a konzultace příslušné tematiky MUDr. Miroslavu Erbrtovi, Kardiologie Na Bulovce, s r.o..

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 18. dubna 2008. Tomáš Kazmar

Contents

List of Figures

List of Algorithms

List of Tables

Abstrakt

Název práce: *Opacity Quantification in Cardiac Angiogram Sequence*
Autor: *Tomáš* KAZMAR
Katedra: *Katedra softwarového inženýrství*
Vedoucí diplomové práce: *Dr. Ing. Jan* KYBIC, *Katedra kybernetiky, FEL ČVUT*
e-mail vedoucího: kybic@cmp.felk.cvut.cz
Abstrakt: Angiografie je základní vyšetřovací metodou u pacientů s infarktem myokardu. Běžně se používá k detekci stenózy v jedné z arterií, data lze ovšem využít i pro odhalení problematických částí kapilár. Takové dvojí použití dat je vhodné hned z několika důvodů. Jednak proto, že katetrizace je invazivní metodou s nezanedbatelným rizikem pro pacienta a především proto, že možnost získání informace o průchodnosti kapilár znamená významný přínos pro rozhodování o následujícím léčebném postupu, zejména o výběru farmak. Stále otevřeným problémem je otázka kvantifikace opacity na škále MBG.[1]

V rámci diplomové práce by se student měl zaměřit na následující:

- shrnout a porovnat jednotlivé metody vhodné pro řešení problému kvantifikace opacity, především registraci obrazu a korekci pohybu
- navrhnout metodu automatické kvantifikace opacity a následně ji implementovat

Klíčová slova: *zpracování obrazu, rentgenová angiografie, registrace*

[1] MBG — myocardial blush grade.

Abstract

Title: *Opacity Quantification in Cardiac Angiogram Sequence*
Author: *Tomáš* KAZMAR
Katedra: *Department of Software Engineering*
Supervisor: *Dr. Ing. Jan* KYBIC, *Department of Cybernetics, FEE CTU*
Supervisor's email address: *kybic@cmp.felk.cvut.cz*
Abstract: Angiography is an important method to diagnose patients with myocardial thrombosis. It is usualy used to detect a stenosis in one of arteries but the data can be further processed to discover problematic capilary areas. This is strongly suggested because the catheterization is invasive, it involves a certain risk for pacient and especially it would be invaluable for further decisions about proper pharmaceutical treatment. Problem of opacity quantification on a MBG scale has not been satisfactorily solved yet.[1]

The student shall discuss the following in his diploma thesis:

- recapitulation and comparisons of various methods applicable to the problem of opacity quantification, especially image registration and movement correction
- develop a method to automatically quantify the opacity and further implement it as well

Keywords: *image processing, X-ray angiography, registration*

Chapter 1

Introduction

In this thesis we try to solve a problem of opacity quantification in angiogram sequence data. As far as we know, there are no published methods targeting this problem and the area was only scarcely studied. In subsequent chapters we first explain why it is important to automate the process of opacity assessment and what is its place in medical imaging. We propose various alternatives of opacity quantification method and their prerequisites. We list advantages of some methods as well as why several approaches, although they might seem promising, do not lead to good results.

In the first chapter we describe a medical perspective, circumstances leading to the treatment and the actual procedure and discuss how the opacity is measured. Afterwards follows a brief demonstration of input data.

In the next three chapters we analyse three topics encountered throughout the method development — vessel extraction, image registration and opacity quantification. Each of these chapters begins with a survey of prior-art methods, with rationale of choosing one method over another and actual description of the chosen methods and results obtained.

In the last chapter there are some technical details about the implementation, the issues encountered and comments about the developed user interface.

Chapter 2

Angiography

In this chapter, we explain the essential medical background, how the data is acquired and how we plan to use it.

2.1 Medical background

Angiogram opacity quantification is a recent method which started to evolve during 90's [89]. The method is based upon the fact that contrast agent washes out differently from normal heart and pathological heart. This is the case also with acute myocardial infarction.

Acute myocardial infarction starts by sudden and usually total athero-thrombotic *occlusion* which leads to *ischemia*[1] in the area distal to the occlusion. The ischemia can cause partly reversible or even irreversible changes to the *micro-circulation*.[2] After the occlusion is removed (either spontaneously or after a treatment) and blood flow in the distal part of artery is restored, it is possible to spot retention of contrast agent in capillaries marked by darker intensities in angiogram. This is referred to as *opacification*. Opacification correlates with *microvascular dysfunction*, that is with the level of damage done to the arterial bed distal to the occlusion [38].

The actual treatment begins when a patient is received with acute myocardial infarction. He is transported to the operating theatre where a physician localises the occlusion of an artery using diagnostic *catheterisation*.[3] Physician selects the artery based on 12-leads ECG performed in an ambulance. When the occlusion is found, the artery is reopened by an intracoronary wire guided through a catheter. The place of occlusion can be predilated by a balloon catheter and the arterial lumen is restored to its normal size by stenting[4] [67]. So, catheterisation which is an invasive method that implies certain risk to the patient is used in both, the diagnosis and the treatment [47, 67].

After the treatment a physician visually assesses whether it was successful, that is he examines the blood flow and micro-circulation from the angiographic data

[1] Insufficient oxygen supply to the heart muscle.
[2] Blood flow in capillaries
[3] Insertion of catheter into an artery, to be explained in the next section.
[4] Placing a stent (wired tube) inside the artery to hold its walls.

obtained during catheterisation. About one year later new data is acquired in a less interventional modality such as scintigraphy. A comparison between angiographic and new data helps the physician decide about medication.

2.1.1 Coronary angiography

Generally, *angiography* enables us to obtain visual information about anatomy of vessel tree and blood flow in it. In coronary angiography, it is necessary to manipulate a catheter through a peripheral artery, eg. femoral artery, all the way to heart. Then a radioactive agent is periodically injected into one or more branches of cardiac arteries. The resulting opacification of arterial tree is recorded. This procedure is referred to as *catheterisation* and the steps are as follows [67]:

1. physician inserts catheter into aorta into an artery in an arm or thigh
2. he pushes catheter up the artery and from aorta he steers it into an origin of the artery to be examined
3. dye is injected through the catheter
4. X-ray sequence is taken during
   (a) *arterial phase* when dye gradually fills and leaves the artery
   (b) *capillary phase* when dye enters micro-vasculature
   (c) *venous phase* when dye leaves through veins

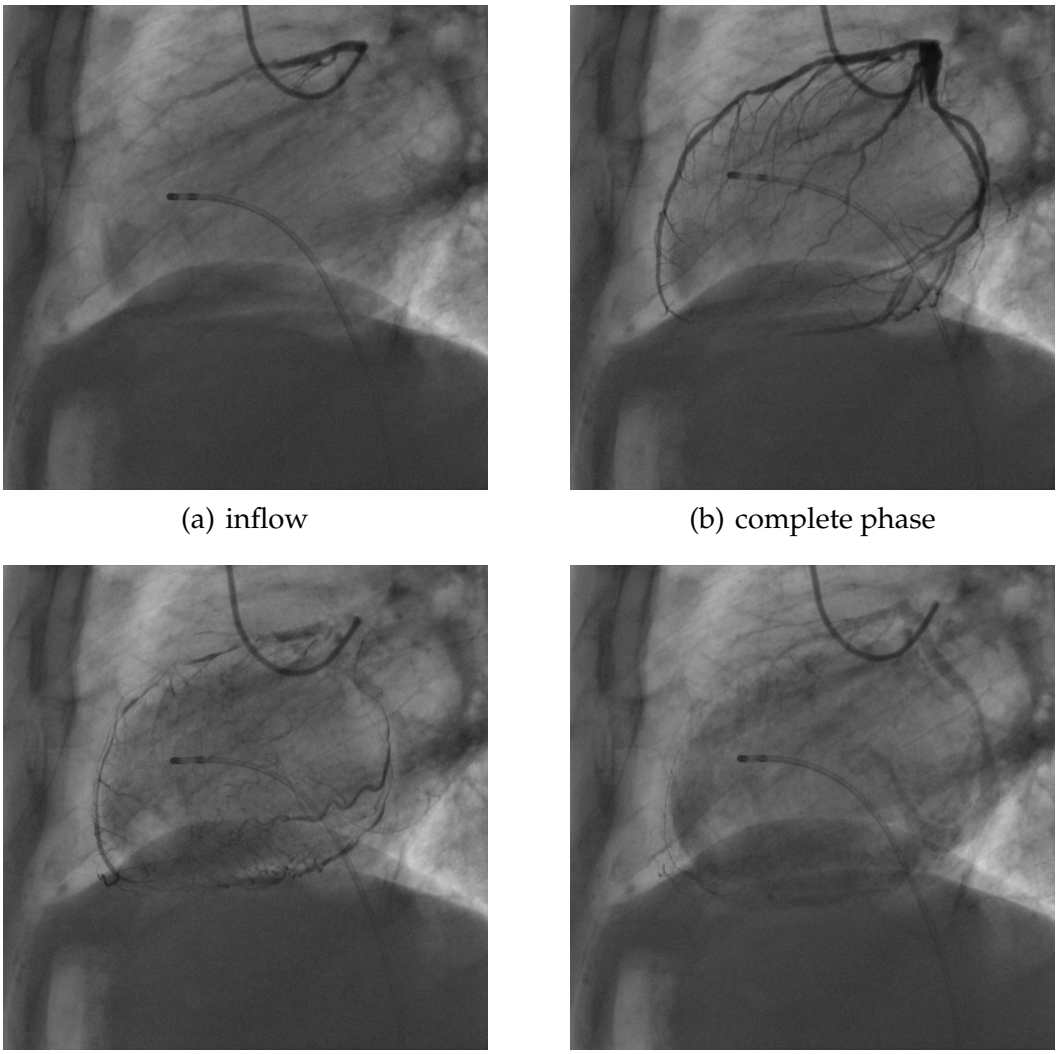
Data acquisition is usually stopped during the capillary phase to limit a radiation dose. Other phases that can be recognised are *inflow* when artery fills with dye, *complete state phase* when artery stays opacified and complete arterial tree is visible, and *washout* when dye clears from vasculature, see Figure 2.2 and Figure 2.1.

Catheterisation as described here, that is recording video sequences of arterial beds is referred to as *arteriography*, or implying that we are imaging coronary arterial beds, *coronarography*, or to emphasise we choose one of arteries, *selective cardiac angiography*. There are also other types differing by physical modality (CT, MR, radiography, fluoroscopy) and imaged part of body (brain, heart, lungs, retina, etc.). Angiograms of different modalities can differ greatly in imaging quality. In our case we deal with single-plane X-Ray selective cardiac angiography. Among angiographic techniques, X-ray angiography still plays the role of a golden standard [67], although it is the oldest modality[5] [89].

2.1.2 Angiographic projections

Different views are defined depending on which artery is to be examined. In the case of a right coronary artery (*RCA*), there are four, and in the case of the left coronary artery

[5]First angiogram was taken by Dr. Sones in 1953 when he accidentally inserted catheter into one of the arteries and the patient surprisingly survived.



(a) inflow (b) complete phase

(c) washout (d) venous phase

Figure 2.1: Examples of phases (L90), we can see gradual filling of vessels with contrast agent (a), till whole arterial tree is visible (b) and contrast agent starts to clear into capillaries (c), at last we can see the vein going from catheter down to electrode in the right part of image, and flushing opacified blood into a chamber (d).

(*LCA*), five views. All of them are derived from standard antero-posterior projection (*AP*) by two rotations, namely pitch and yaw, to obtain a more appropriate projection. This means all views go from patient's chest backward and a projection name describes the position of an X-ray head. In left views X-ray head is positioned left of the patient and looks to the right, in right from right to left. *Caudal* stands for downward and *cranial* for upward direction.

Projections are either left or right anterior oblique, namely RCA projections are *LAO 60*, *LAO/Cranial*, *RAO/Cranial*, *RAO/Caudal*, and *LCA* projections are *RAO/Caudal*, *RAO/Cranial*, *LAO/Cranial*, *LAO/Caudal* (so called spider view) and *LAO 90*. Data taken from one view at one time is called a *run* or *sequence*. In a complete procedure there

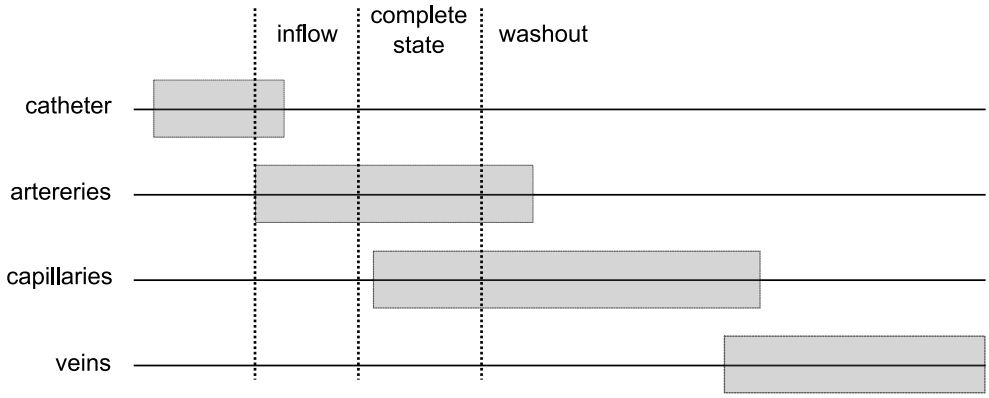


Figure 2.2: Layout of phases in respect to contrast agent which enters arteries through catheter after an injection, then blood perfusion takes it through capillaries into tissue and back into veins.

is at least one run for each of all nine views. The reason to take a run of healthy artery is for a comparison with the one under examination but the concrete selection of projections differs from patient to patient and is up to the performing physician to decide which projections are apt for given situation. The same goes for exact positioning of projections which can also vary.

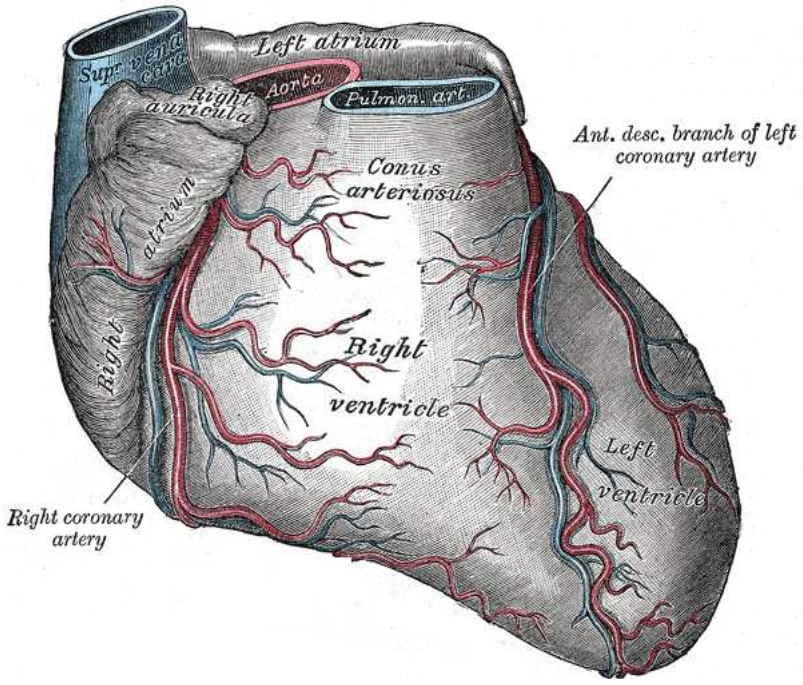


Figure 2.3: Surface of the heart — RCA (left), LCA (right). Taken from [39].

2.2 TIMI and TMP Scales

Blood perfusion in coronary arteries is measured using two scales: TIMI flow and TMP. TIMI flow scale [101], gives information about how quickly an artery bed is filled with new blood. TIMI scale is defined as follows:

TIMI0 — no perfusion (no flow beyond the point of occlusion)

TIMI1 — penetration without perfusion (dye passes beyond the obstruction but fails to opacify the entire coronary bed distal to the point of occlusion)

TIMI2 — partial perfusion (dye slowly opacifies the coronary artery)

TIMI3 — complete perfusion (dye opacifies bed distal to the obstruction as promptly as the bed proximal to the obstruction)

TMP scale defined in [38] describes the rate of perfusion of microvasculature — the micro-circulation. It is also called MBG (Myocardial Blush Grade). It is defined as:

TMP0 — failure of the dye to enter microvasculature.

TMP1 — dye slowly enters the microvasculature but fails to exit, dye staining is present at the time of the next injection (approx. 30 seconds).

TMP2 — delayed entry and exit of dye from microvasculature, dye is strongly persistent after 3 cardiac cycles of the washout phase, its intensity diminishes minimally.

TMP3 — normal entry and exit of dye from microvasculature, dye is gone or is mildly persistent after 3 cardiac cycles of the washout phase.

2.3 Input data

Data used in this thesis was obtained with an intention to treat at Hospital Na Bulovce, Prague, Czech Republic, on an Philips Integris H machine. A total of six complete studies was available for this thesis. Patients imaged in the medical study all had TIMI values of at least TIMI1 flow grade after the re-perfusion. Acquired angiographic data was saved onto CDs in a *Digital Communications in Medicine Standard (DICOM)*.

DICOM format is one of formats used for storage of medical data [7]. In the root directory, there is a `dicomdir` file pointing to actual graphic data which are stored in a separate directory. A DICOM file itself includes metadata specifying the modality, information about the image data, such as the number of bits per pixel, intensity mapping, extent of the valid region, and also data about the patient. Because of the latter DICOM data files should be anonymised (personal information removed) unless it is legal not to do so.

In our case, each file covers one run which is composed of a sequence of frames, encoded as 512x512 JPEG images with lossless compression (non-hierarchical, first order prediction). Sequences usually consist of 60 to 120 frames, with time step of 80

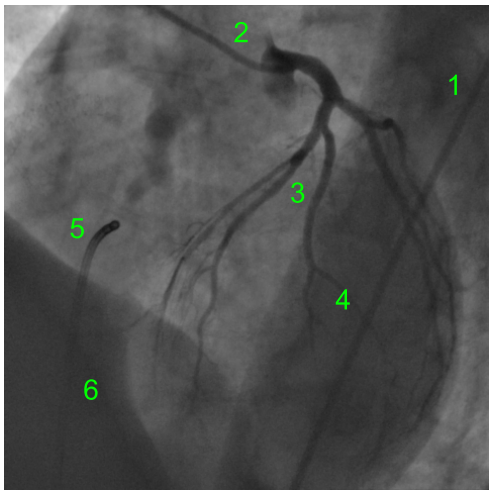


Figure 2.4: Angiogram. Catheter going upwards (1), ending in a mouth of the artery (2). Part of the ribcage (3) is only faintly visible branching from the spine (4). In the left bottom corner you can see an electrode (5), and also a diaphragm (6). Sometimes liver is seen and can be mistaken for the diaphragm as they look similar. During angioplasty there is guide wire and sewing wire, as well. One of the factors causing noise is a level of fat in the patient.

milliseconds. Images are 8bit grayscale with `MONOCHROME2` photometric interpretation where black is the highest opacity, white the lowest. For an example see Figure 2.4. With given time resolution there are 12.5 frames per second which corresponds to about 6.25 to 12.5 frames per heart beat.

Only the inner 480x480 area of frames is actually carrying data. This blanking of image borders is caused by *physical collimation* [108] and its purpose is to reduce the dose patient receives as the physician is interested only in a small portion of the vasculature. To achieve this, four lead sheets, one from each side is inserted into the field of view. Position of these plaques is saved in proprietary Philips DICOM tags (`0019|1002`, `0019|1004`, `0019|1006`, `0019|1008`) and the shape in tag `0019|1000` which is in our case `RECTANGULAR`.

Chapter 3

Overview

3.1 Problem definition

This work is meant as a proof of concept. It's aim is to help the physician in evaluation of the contrast agent perfusion from X-ray angiographic data using TMP grades[1] and to verify whether it is possible to quantify the perfusion automatically. Automatic evaluation has the advantage of being objective and repeatable measure while standard TMP grade evaluation suffers from inter-observer variability and even intra-observer variability (due to fatigue etc.) [116, 105].

We are interested in the evaluation of contrast agent perfusion of capillaries from coronary angiographic data. We look for opacified patches next to the artery to be examined. These patches are formed by capillaries which stem from main arteries and go inside of the heart wall. Individual capillaries are too small to appear in the images, or appear only faintly, even in complete state frames. Abnormalities in the opacification signal the level of injury. Unfortunately, changes in intensity between adjacent frames are subtle and it is hard to perceive them, even for a skilled person. This means that data from larger part of a sequence have to be used.

An ideal automatic method should be able to tell a TMP grade given an angiographic sequence and a region of interest (ROI). Desired outputs are opacity enhanced sequences, or time-series of opacity in a ROI. The method should include a registration to accomplish motion compensation. Another possible outputs can be a still image of the heart with evolving opacity or with intensity-encoded opacity values.

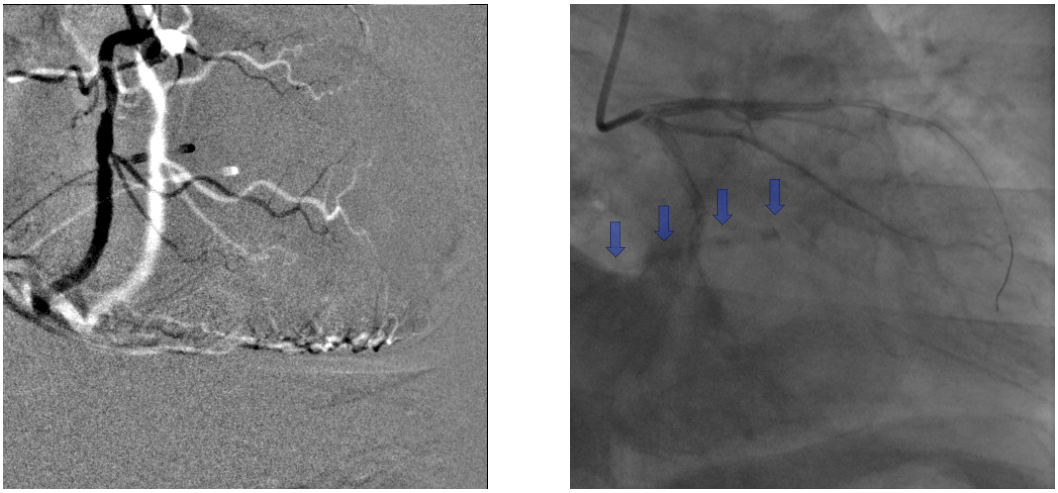
3.1.1 Prior-art

At the time of writing this thesis we are not aware of any other published method that addresses the problem of automatic opacity quantification from X-ray angiographic data, save the work by Condurache et al. who studied tracking of a predefined region in angiogram sequence during complete state phase [25, 2]. More in-depth discussion of this method is provided in Chapter 6.

[1] For definition of TMP scale see Section 2.2

3.1.2 Data properties

There is a relatively high level of noise. The images contain few clearly distinguishable features which is quite typical for the X-Ray modality. A significant part of the scene (eg. diaphragm, or ribcage) is irrelevant for our purposes, although it is clearly apparent in the frame. Sampling frequency is relatively low and sometimes objects in the image move in different directions and speeds and even consecutive frames can differ a lot (up to 30 pixels in position of arteries), see Figures 3.1(a) and 3.1(b).



(a) (b)

Figure 3.1: Input data examples. Difference frame of two consecutive frames in RAO/Caudal projection (a). The two artery beds compared in images are shifted, intensity values are high as arteries are strongly opacified. Frames taken during a systole may be a blurred due to movement (b), a catheter moves so quickly that it is hardly visible at all.

A complication with our input data is that intensity histograms of individual frames have very short support, see Figure 3.2, and differences of intensity values in smaller local regions are often as low as 10 per 256 levels. In some cases, the physicians can probably figure out the opacification from normalised frames where these differences are nearly twice as big. Of course, such normalisation increases the noise as well.

Figure 3.2: Histogram values in whole sequence, time runs from top to bottom, intensity values from left to right. Values have been normalised.

3.2 Method outline

We shall outline the proposed algorithm. Considering the input data, we identify the following steps needed to process the angiographic sequence:

1. preprocessing
   (a) de-noising
   (b) crop collimated borders (Section 2.3)
   (c) estimate heart rate (Section 6.2.1)
2. motion compensation (Chapter 5)
   (a) rigid registration
   (b) precise registration
3. opacity enhancement (Section 6.2.1)
4. quantification (Section 6.2.2)
   (a) define ROI throughout the sequence
   (b) compute opacity curves (time series of opacity)
   (c) applying a model to opacity curves (parameter identification)

Rigid registration (item 2a) is expected to compensate for camera movements and partially for unintended breathing of a patient present in some sequences. With precise registration (item 2b) we investigate the different registration approaches, a feature-based registration which uses vessel center-lines produced by vessel-extraction methods (Chapter 4), and an image-based approach that works directly with pixel intensities. In both cases, motion compensation should use elastic registration to compensate movement of arteries over a cardiac cycle. Another problem is that there are lot of objects in the image that can mislead the registration process. Moreover, we have to deal with effects inherent to a 2D projection of 3D objects, occlusion between objects, self-occlusion, change in mutual position of objects due to off-plane rotation etc. For example, a strong feature might be unusable because it is caused by a mutual occlusion of collaterals emanating from the artery which move differently. Registration has to be regularised to obtain a suitable estimate of the displacement field even away from the main arteries where the capillaries are.

The purpose of the enhancement step (item 3) is to simplify the evaluation of opacity changes. In Figure 3.1(a) we show a difference of two consecutive frames during complete state phase, so that whole arterial beds are clearly visible.[2] We call such a frame a *difference frame*, and a sequence obtained by such subtraction a *difference sequence*. By a proper selection of an aligned reference frame, or estimated background of the frame we can increase the intensity of opacity. Unless an alignment of individual

[2]For explanation of angiographic phases see Section 2.1.1.

frames is accomplished, we get subtraction artifacts as illustrated in Figure 3.1(a). Motion compensation (item 2) is thus essential.

To extract opacity curves (item 4b), first the ROI must be entered (item 4a). As automatic definition of the ROI is difficult, we let the user interactively set the ROI in one or several frames which is then tracked to all remaining frames using recovered deformations. Next, we apply a model to the extracted opacity curve to estimate opacification parameters.

Chapter 4

Vessel extraction

Detecting a vessel tree is crucial for many applications, including angiographic images. To understand and analyse the data we need to identify the vessels. The exact sense of what identification is can differ, we shall calculate a measure of vesselness for each pixel, and alternatively detect a vessel centre-line. The former is useful in quantification weighing based on this measure, while the latter is aimed at feature-based registration.

4.1 Introduction to vessel extraction

Vessel extraction techniques have been studied intensively during last two decades [96, 70]. One of the main requirements is to detect vessels quickly, or/and in a large numbers of images. Some modalities do not provide good contrast, and in general the background is not homogeneous, such as in our case. Many of the methods work not only with 2D data but also treat more complex cases like time-varying sequences or volumes.

Vessel extraction is a rather broad concept. The methods fall into three groups:

1. vessel-enhancing approaches,
2. approaches that find a vessel centre-line and optionally some approximation of vessel width, and finally
3. approaches that deliver complete segmentation of the vessel tree

Some centre-line extracting approaches furthermore construct a graph of parts of the skeleton that in some way represents interconnections between vessels. These graphs are later labelled and used either to search correspondences from multiple views or some kind of image understanding is done. There is a wide variety of methods solving this [45], [56], [110] or [17].

When extracting vessels there is some apriori information we can use. First, vessels are relatively thin elongated structures. Second assumption is about the intensity (in our case vessels are seen as dark on bright background). More assumptions can be made about the orthogonal intensity profile of a vessel. Rather crude approximation is

to model the vessel as two opposing parallel step edges, more common is a Gaussian-like profile [80]. These assumptions have to be further refined to be useful in actual methods that can differentiate between a vessel and non-vessel locations.

Vessel extraction is based on ridge detection, region growing or various artificial intelligence techniques ranging from neural networks to particle swarm optimisation. The following classification is loosely based upon a classification proposed by Kirbas and Quek in their interesting survey on the topic [53]:

1. pattern recognition techniques
   (a) matched filters
   (b) ridge-based or skeleton-based
   (c) region growing
2. model-based approaches
   (a) deformable models
   (b) parametric models
3. tracking-based approaches
4. artificial intelligence approaches

Methods usually fall into more than one category. If a method provides just an enhancement and we need a skeleton or segmentation we can always add thresholding, although this increases the number of parameters to tune.

In X-ray angiography, we can detect vessels because of opacity, and so we should prefer complete state images where vessels are opacified. In the inflow phase only the central part of the vessels is visible as a dye carried by blood flows into central part quicker. On the other hand during washout the central part is cleared earlier than the borders of lumen where blood flows more slowly.

Regarding width of vessels, thick vessels are detected more precisely and quite easily, whereas thin vessels are either only faintly visible or even not visible at all and their recovered width is often biased [94].

4.2 Survey of vessel extraction techniques

For the lack of space, we will describe only a few selected methods and refer the reader to an exhaustive review [53]. We restrict our discussion to methods based on region growing, ridge-based, matched filters and deformable models because we regard them as particularly interesting. We also select these categories as methods we implemented fit into them, namely these are a method by Poli [80], Efstratiadis [46], Frangi [35], and Guo [42].

4.2.1 Background subtraction

The simplest approach is a non-linear filtering, subtracting a background estimated using a median filter:

$$B(x, y) = I(x, y) - m\left(N(x, y)\right)$$

where m is median of a kxk neighbourhood of (x, y), denoted $N(x, y)$. It is commonly used as a preprocessing step by other methods [42],[2]. To extract background we apply a median filter with a large kernel which selects non-vessel part of image. A top-hat filter can be used instead of a median (for black vessels). After the subtraction, both variants yield images with suppressed background. The size of the filter must be selected carefully not to remove large vessels, so it should be larger than the largest vessel we want to detect. In our case best filter sizes ranged from 31 to 51 pixels, depending on input data. The result maybe usable as a preprocessing method, or as a means to balance the image intensity [42]. Some methods use averaging instead of median filtering [46], although in our experience results with median are slightly better. A typical result can be seen in Figure 4.1.

Figure 4.1: Vessel enhancement using background subtraction (b) of an input (a), for the visualisation purposes we masked out the image by a triangle thresholding method which works well for data with a flat histogram part corresponding to an object [112].

4.2.2 Ridge-based methods

The concept of ridge is inspired by an analogy with terrain, an angiogram is treated as a height map where ridges correspond to vessel centre-lines. For bright vessels on dark background, we define the height map to be equivalent with I. For dark vessels on bright background, we take the inverse of image $1 - i$. Here, we describe three methods by Guo [42], Efstratiadis et al. [46] and Frangi [35]. The height map is either processed in its raw form [42] or based on analysis of low and high eigenvalues of the Hessian [46, 35] where Hessian of the input image I is given by:

$$H(x,y) = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix}$$

To detect dark vessel-like structures on bright background, we should look for one big positive eigenvalue and one comparatively small.

One of ridge-based vessel extraction methods which is quite representative was developed by Guo [41] and later modified to angiographic input [42]. The main idea is to search the height map in the direction of the greatest descent. Preprocessing is essential as the height map must be smooth.

Guo suggests to do intensity balancing and anisotropic smoothing to remove the noise usually found in X-ray images. First, the image is balanced by background subtraction with median as in Section 4.2.1. Next *curvature smoothing* is applied [1]. Curvature smoothing is a type of anisotropic smoothing which tends to preserve vessels as opposed to noise.

Preprocessed image is then locally searched by simulating water falling on our smoothed surface. First, we select the ROI by thresholding the preprocessed image with one global threshold. This can be done as the image is expected to be balanced by preprocessing. From each pixel in ROI we move to its neighbour in the greatest descent direction constructing a graph. We add the original pixel to a list of children of the target pixel. This way we create a stream-in image S_{in} where the value at position $\vec{x}$ is a degree of given pixel in the graph. We select pixels S_{mp} where at least two streams merge, that is where $S_{in}(\vec{x}) \leq 2$, and we classify them into:

- ridge start points R_s — no incoming stream from S_{mp}
- ridge intermediate points R_{ip} — at least one incoming stream from S_{mp} and exactly one outgoing stream, and
- ridge terminating points R_t — at least one incoming stream from S_{mp} and no outgoing stream, and similar
- ridge vanishing points R_v — at least one incoming stream from S_{mp} and an outgoing stream leaving ROI.

R_t and R_v are roots of trees in the constructed graph $G(S_{mp}, E)$ where edges E are the taken steepest descent paths, see Algorithm 1. Complexity of this detection is linear in terms of number of checked pixels as we do constant work at each pixel.

To detect the centre-lines, we have to process G further. Guo proposes a curve relaxation method, first the directions of ROI points in graph G are locally quantised and then saliency reconstruction as introduced by Ullman and Sha'ashua is applied to interconnect the curves [103].

Unfortunately, our implementation did not perform quite as well as what the author claimed. We do not get sufficiently connected candidate pixels for centre-lines to be able to quantise vessel directions effectively and run saliency reconstruction, see Figure 4.2. The main reason seems to be different data we tested it with. Other possible

Algorithm 1 Guo's ridge-based detection

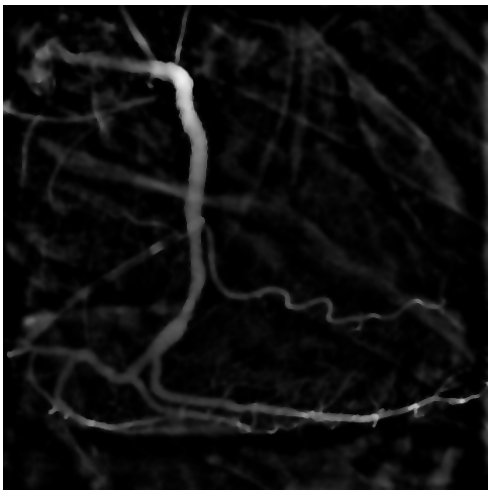
Input: matrix I, constant k
Output: ridge graph $G(V, E)$

$B(x, y) = I(x, y) - \frac{1}{k^2} \sum_{(a,b)\in N(x,y)} I(a, b)$ where $N(x, y)$ is kxk neighbourhood
$B_{smooth} =$ anisotropic smoothing of B
compute matrix $D(x, y)$ of greatest descent directions for each pixel according to B_{smooth}
initialise stream-in matrix $S_{in} = 0$
for all points v in ROI
    $S_{in}(v + D_v) = S_{in}(v + D_v) + 1$
$S_{mp} = \{ v \mid S_{in}(v) > 1 \}$
for all points v in S_{mp}
    if D_v is undefined
        $R_t = R_t \cup \{v\}$
    else
        if a point $v + D_v$ is in ROI
            add v to its children list
        else $R_v = R_v \cup \{v\}$

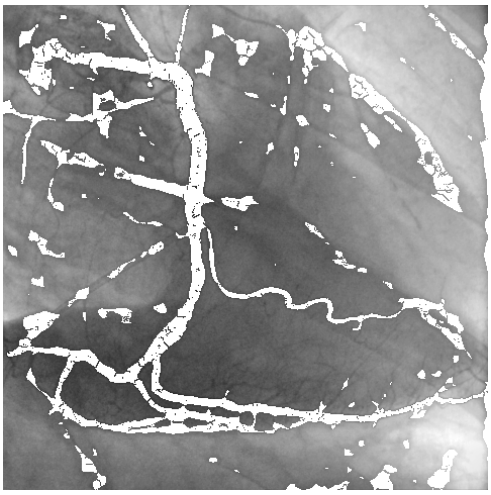
reason can be a different implementation. For example, the author does not precisely specify the preprocessing. We apply Perona-Malik diffusion [79] as an anisotropic filtering step and although we do not expect that preprocessing step is the problem, it can seriously affect the results. As well it can be for subtle errors in description of the algorithm as the author does not clearly describe some aspects of selecting ridge vanishing nodes and ridge children detection.

In Figure 4.2, you can see that stream merge points S_{mp} approximate the vessel tree and that ridge terminations lie at the vessel centre-line. But we were not able to reconstruct the whole centre-lines following Guo's article.

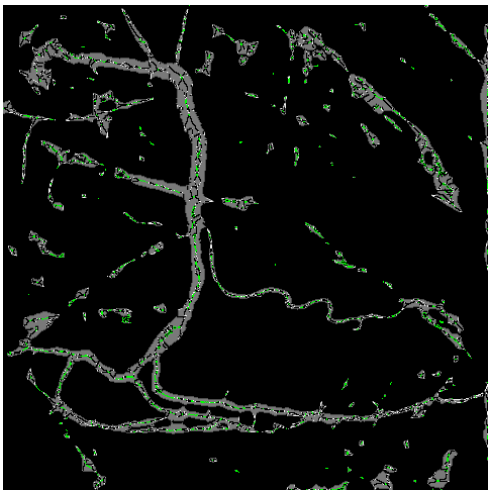
The second ridge detection method that we studied more profoundly is due to Efstratiadis et al.[46]. They compute Hessian H of input image I — partial derivatives I_{xx}, I_{xy}, I_{yx}, I_{yy} are computed by convolution with derivatives of a Gaussian. Then they apply eigenanalysis to all matrices $H(x, y)$ yielding eigenvalues λ_1, λ_2 with corresponding eigenvectors ω_1, ω_2. Next topographic labels are assigned following the rules described in Table 4.1. Conditions that check on zero equality have to be evaluated using zero-crossing detection, as for example $||\nabla I|| = 0$. can be true anywhere inside the pixel, and not precisely on a grid position. Therefore, values of I_x, I_y are interpolated at pixel area boundary in the direction of ω_1, ω_2 — locations p_1,



(a) preprocessed image



(b) overlay of S_{mp}



(c) output labelling

Figure 4.2: Vessel enhancement using Guo's ridge-based method. First image shows contrast balanced, smoothed input (a), second shows stream merge points (b). In the last image (c), we can see the ROI (gray), stream merge points (white) and ridge terminations (green).

q_1 and p_2, q_2. Finally, I_x is considered to be zero inside the pixel if interpolated values $I_x(p_1)$ and $I_x(q_1)$ differ in sign, and similarly for I_y.

Image point is considered a candidate centre-line point if it was labelled either as ridge, saddle or peak. These pixels form an image mask M. Thin elongated regions only a few pixels wide must be removed by post-processing. Efstratiadis et al. propose to first use filtering based on local contrast balanced intensity image B as in Guo's algorithm — subtraction of local neighbourhood's mean:

$$B(x,y) = \begin{cases} I(x,y) - \frac{1}{k^2} \sum_{(a,b)\in N(x,y)} I(a,b) & \text{if } M(x,y) = 1 \\ 0 & \text{otherwise} \end{cases}$$

applied only to pixels belonging to M. Then they threshold B with lower and higher

label	*condition*
peak	$\|\|\nabla I\|\| = 0$, $\lambda_1 < 0$, $\lambda_2 < 0$
pit	$\|\|\nabla I\|\| = 0$, $\lambda_1 > 0$, $\lambda_2 > 0$
saddle	$\|\|\nabla I\|\| = 0$, $\lambda_1\lambda_2 < 0$
ridge	$\|\|\nabla I\|\| = 0$, $\lambda_1 < 0$, $\lambda_2 = 0$, or $\|\|\nabla I\|\| \neq 0$, $\lambda_1 < 0$, $\nabla I\omega_1 = 0$, or analog. for λ_2
ravine	$\|\|\nabla I\|\| = 0$, $\lambda_1 > 0$, $\lambda_2 = 0$, or $\|\|\nabla I\|\| \neq 0$, $\lambda_1 > 0$, $\nabla I\omega_1 = 0$, or analog. for λ_2
flat	$\|\|\nabla I\|\| = 0$, $\lambda_1 = 0$, $\lambda_2 = 0$
hillside	otherwise

Table 4.1: Topographic labels used in ridge detection.

thresholds, T_l and T_H. The resulting images are B_l and B_h. Next they extract connected components of B_h and remove small components, let us call the result C. Finally, C and B_l are subject to *geodesic reconstruction* where C is a marker and B_l is a mask [107]. One step of geodesic reconstruction is morphological dilation followed by applying a mask:

$$C = (C \oplus E)B_l$$

where E is a structuring element. This step is repeated until convergence.

The same authors further improved this method using morphology operations of directional opening and closing applied before topographic labelling [44].

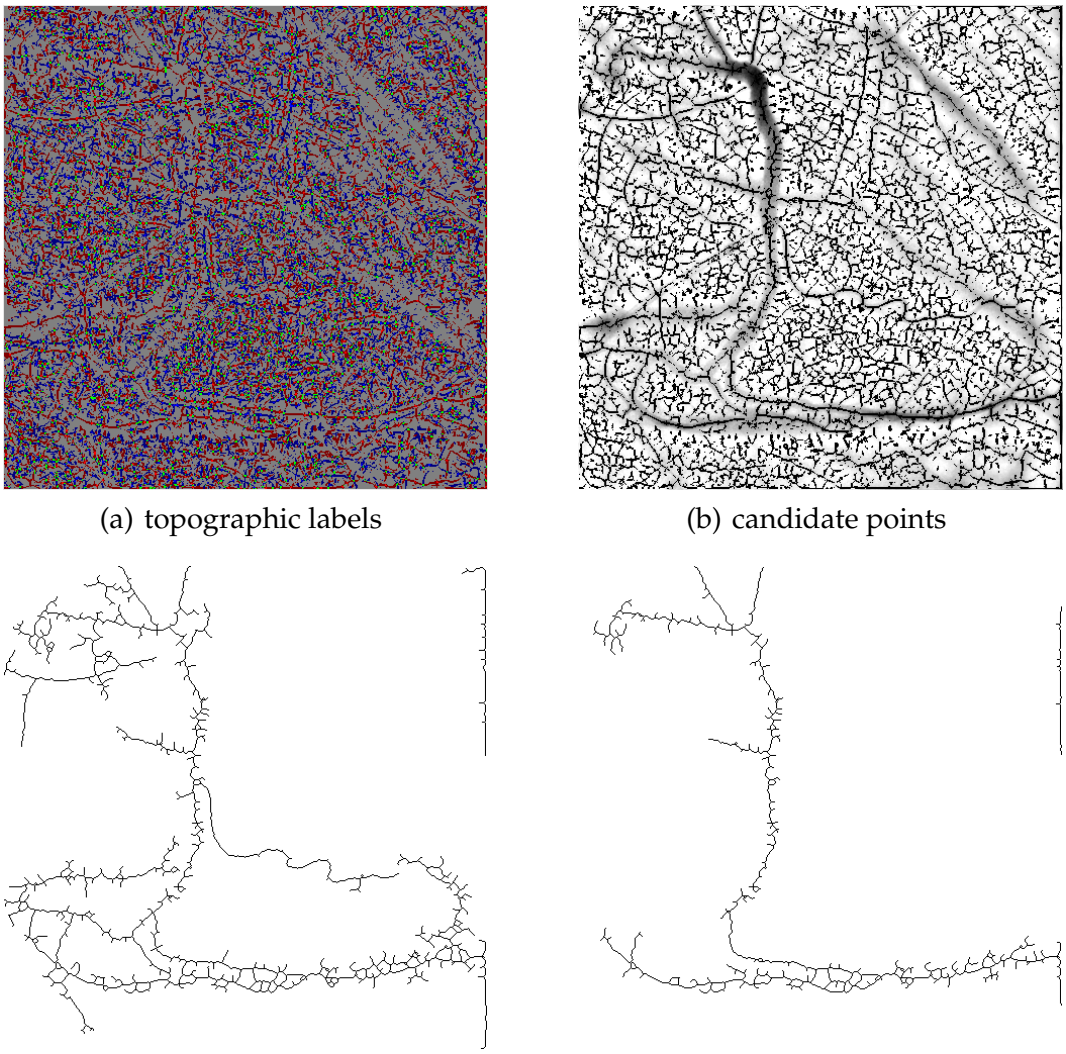
Our implementation follows the description given in a very concise way in the article [46]. We implemented zero-crossing search for topographic labelling using linear interpolation. The post-processing uses geodesic reconstruction by dilation with 3x3 structuring element.

The method performs quite well, however if you look at the Figure 4.3, you can see that the method produces disjoint parts. We finally tested the method with intensity thresholds set to $T_h = 0.2$, $T_l = 0.1$ and components size threshold $T_c = 200$. Unfortunately, automatic threshold setting is difficult.

We had to adjust the algorithm to work with our data, as the original algorithm fails because lower thresholded image B_l is not continuous. To prevent it we include dilating B_l. However, the main disadvantage of this method for our purposes is computational complexity of labelling and post-processing using morphology as well, see a comparison Table 4.4 in Section 4.4. Both could be implemented more efficiently but, as there are difficulties with threshold selection anyway, and the performance is not excellent, it does not seem worthwhile.

Another method [35] is directly based upon an analysis of a Hessian H and its eigenvalues throughout the image. We formulate it for the whole sequence to emphasise that contained normalisation parameters should be constant for all frames.

For dark vessels local analysis of low eigenvalue λ_l and high eigenvalue λ_h, λ_l,



(a) topographic labels

(b) candidate points

(c) reconstructed ($T_c = 100$, $T_l = 0.15$, $T_h = 0.25$)

(d) reconstructed ($T_c = 150$, $T_l = 0.20$, $T_h = 0.25$)

Figure 4.3: Vessel centre-line detection using Efstratiadis' method. Intermediate results (a-b) and two results of post-processing with different settings (c-d). Topographic labels are peak (red), pit (blue), saddle (green), ridge (dark red), ravine (dark blue), flat (light gray), hillside (dark gray).

where $|\lambda_l| < |\lambda_h|$ should give:

$$\begin{aligned} \lambda_h &\gg \lambda_l \\ \lambda_l &\approx 0 \\ \lambda_h &> 0 \end{aligned}$$

The first two conditions signal vessel-like structure and the last one means that it is dark. Authors propose to use two criteria S and R_B. R_B is a ratio of blob-likeness:

$$R_B = \frac{\lambda_l}{\lambda_h}$$

and will be small when first or second conditions are met. Moreover, it remains

bounded by $-1{,}1$ as $|\lambda_l| < |\lambda_h|$. Second criterion S accounts for structureness and is defined as the Frobenius norm of the Hessian:

$$S = ||H||_F = \sqrt{\lambda_l + \lambda_h}$$

It should be small for noisy areas with little structure. The actual vesselness function g is then defined as:

$$g_{c,\beta}(\lambda_l, \lambda_h) = \begin{cases} \left(e^{-R_B^2/2\beta^2}\right)\left(1 - e^{-S^2/2c^2}\right) & \text{if } \lambda_h > 0 \\ 0 & \text{otherwise} \end{cases}$$

Parameters c and β act as normalisation, authors fix β to 0.5 and propose to set c automatically as a half of the maximum Hessian norm.

Algorithm 2 Frangi's ridge-based detection

Input: sequence
Output: vesselness $v(f, x, y)$

set parameters c and β to appropriate values
for all frames f
  for all positions x, y
    calculate Hessian $H(x, y)$ using derivatives of Gaussian
    eigenvalue analysis of $H(x, y)$ yielding λ_h and λ_l
    evaluation of $v(f, x, y) = g_{c,\beta}(\lambda_l, \lambda_h)$

Algorithm 2 as formulated above can reliably detect vessels of a single width corresponding to the size of the filter used to calculate the Hessian. To achieve invariance with respect to scale this algorithm is turned into a multi-scale one. We apply the single-scale algorithm with different sizes of the Hessian filter and then add the results together. This is a classical multi-scale approach studied in detail by Lindeberg who formulated the so-called scale-space theory [59]. According to Lindeberg, if we calculate derivatives at scale s by convolution with the derivatives of Gaussian:

$$G(x, s) = \frac{1}{2\Pi s^2} e^{-\frac{||x||}{2s^2}}$$

we have to normalise them by a factor of s^γ. When no scale is to be preferred, γ is set to 1. Multi-scale version (Algorithm 3), selects a maximum response across all scales as:

$$v(f, x, y) = \max_s v_s(f, x, y)$$

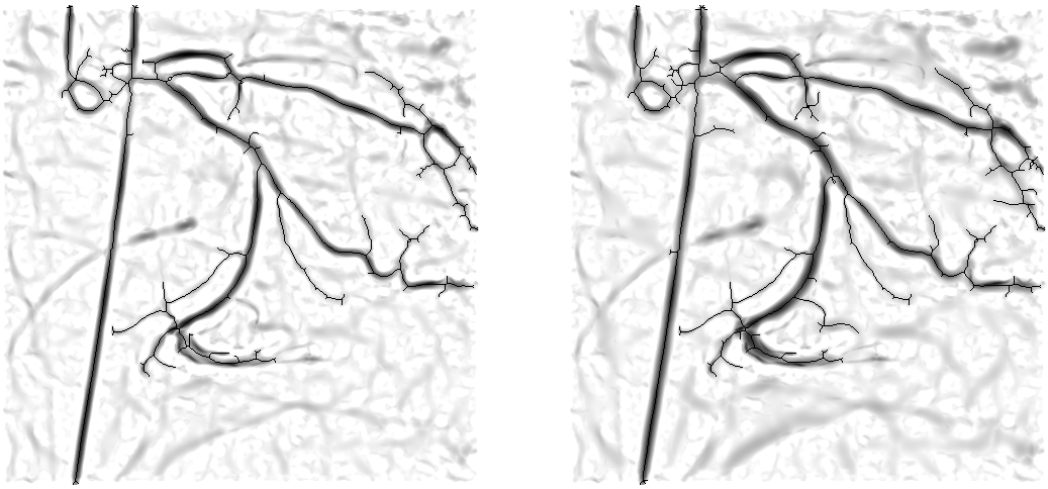
We compute the Hessian by a recursive Gaussian at a given scale which includes scale normalisation [29]. Frangi's algorithm provides a vesselness measure (Figure 4.4), so when there is a need for centre-line we have to post-process the result. We apply thresholding with hysteresis to the vesselness and compute a skeleton of the resulting mask.

Algorithm 3 Multiscale sequence vesselness filter

Input: sequence
Output: vesselness $v(f, x, y)$

for all scales s

use vesselness filter to obtain $v_s(f, x, y)$ (Algorithm 2) using normalised Hessian $H_s(x, y)$
$v(f, x, y) = max_s v_s(f, x, y)$



(a) Frangi 2,4 (b) Frangi 2,4,6

Figure 4.4: Vessel extraction using Frangi's algorithm — vesselness along with centre-lines. For visualisation purposes the vesselness is inverted.

Special care must be taken to properly setup the border conditions when implementing the convolution for Hessian computation. Considering the vessels are dark, the best way to pad image is by zero. On the other hand errors arise when using constant border padding and there is a light line at the border. In this situation we get high responses next to the border and so a spurious vessels near the border.

Related approach to vessel enhancement is due to Maniesing et al [63]. The authors use anisotropic diffusion based on local vesselness measure from Hessian analysis to enhance the appearance of vessels. In the article they claim it to be superior to other diffusion methods used to enhance brain images containing vessels.

4.2.3 Region growing methods

Region growing methods are based on the assumption that vessels are interconnected areas that share some common property. A typical algorithm first selects some starting points, *seeds*, and continues to add further neighbouring pixels until some stopping

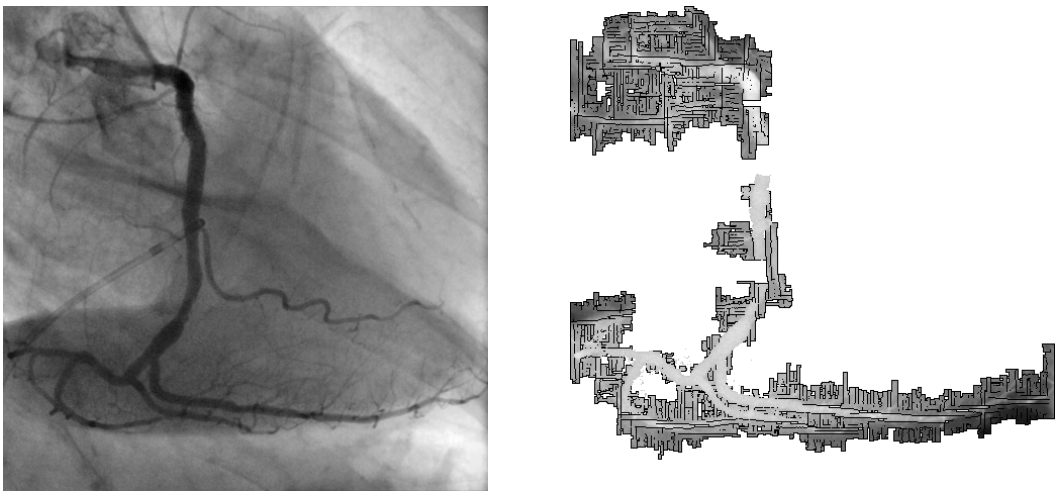
condition is met or no more pixels can be added.

A typical example of region growing is an algorithm by Kirbas and Quek [83]. The algorithm is based upon the idea of wave propagation through a medium of varying density. Wave is propagated to all directions, alternating 4-neighbourhood with 8-neighbourhood to achieve more circular-like spreading of the wave. Quantised image intensity serves to determine the propagation speed in the form of medium density. A sigmoid model is used to define quantisation levels, levels above a certain threshold are marked as boundary with zero propagation speed. For each pixel, there is a variable holding its medium state which is initialised to medium density. The medium state is iteratively decreased in the pixels neighbouring the wave. When medium state reaches zero, pixel is assigned the current wave state. After the propagation is finished, we have obtained a distance function with respect to the medium density. Traceback is then performed in the steepest ascent direction of wave states to find the vessel line.

In some sense this method is comparable to level set and fast marching methods by Osher and Sethian with the important distinction of being purely integer based and so being quicker and easily paralelisable [83]. Result of this method is the vessel connectivity, not the segmentation itself.

Another approach that can also be classified as region growing is by O'Brien and Ezquerra [71]. They let the user select the seed points and grow the region estimating the vessel width and computing a threshold of intensity from local neighbourhood to decide whether the point is added. This method can also be viewed as vessel tracking.

The method is supposed to work for bi-plane X-ray angiographic sequence data which is close to our input data, however, we encountered several problems implementing it as it is not clear how to achieve vessel width estimation needed for point classification which is a crucial part of the algorithm. Leaks outside the vessel are too frequent, see Figure 4.5.



(a) original frame (b) processed

Figure 4.5: Vessel segmentation using O'Brien and Ezquerra's region growing method. The most important disadvantage of the method are the leaks outside the vessels (b). Border pixels are black, unvisited background pixels are masked out.

In some aspect the approach of Efstratiadis et al. described in Section 4.2.2 can be labelled as region growing as well, because of the last step involving geodesic reconstruction.

4.2.4 Matched filters

We presume the intensity profile of a vessel to be specific in some way. This enables us to design a filter that will match this profile and then apply convolution with this filter to image locations we want to examine. This is referred to as the *matched filter*[1] approach [53]. One special class of matched filters was coined as *steerable filters* by Freeman et al. in [36], these are basically matched filters with variable orientation. This way we can have a filter tuned to supposed vessel profile with few plausible orientations, apply all of them to the image and combine the results. This has the advantage of an easy implementation, on the other hand it is quite computationally intensive as we convolve with large filters.

An example of a vessel extraction algorithm that uses matched filters is that by Poli and Valli [80]. It successfully solves the computational load problem and authors claimed back in 1997 that it was nearly apt for real-time vessel extraction. The basic idea is to decompose matched filter into a few simple masks. The authors take a bar-like Gaussian mask and by a top-down approach decompose it into simple operations such as summing up two neighbouring pixels in a row or column.

First a minor optimisation of image smoothing with large Gaussian kernels is presented, by approximating Gaussian kernels with binomial kernels and decomposing these to uniform kernels. In such a way a smoothed image can be computed using only $8\sigma^2$ additions of adjacent pixels and bit shifts. (From current point of view it is not that interesting.) Directional derivative I_n of an image I can be approximated by convolution with the derivatives of Gaussian:

$$I_n(x,y) = \frac{dI(x,y)}{dn} \approx (n_1 G_x(x,y) + n_2 G_y(x,y)) * I(x,y)$$

if we precompute two directional derivatives $G_x * I$, $G_y * I$ we can get an approximation of directional derivative I_n using one addition and two multiplications only.

Bar-like masks proposed in the article are built from:

$$B_n(x,w) = G_n(x+wn) - G_n(x-wn)$$

representing two opposite edges, $2w$ away from each other in direction n. If we sum $2N$ such masks distributed over $2l$ length along the desired vessel direction t orthogonal to n, we get the bar-like mask to detect vessels:

$$B_n(x,w,l) = \frac{1}{2n+1} \sum_{k=-N}^{N} B_n(x + l\frac{k}{N}t, w)$$

Outputs of $B_n(x,w,l)$ for varying n are calculated and a maximum is used:

$$B(x,w,l) = \max_n B_n(x,w,l)$$

[1] Sometimes it is called a matching filter.

Further, authors propose an improvement, called *validation* which attenuates vesselness if the current point cannot be validated [80]. This is achieved by comparing intensities in direction normal to the current filter, if the current point is a vessel point it must have lowest intensity along a normal.[2]

To increase the range of vessel radii to be detected classical multi-scale approach is used as we take maximum output of filters with various sizes:

$$V(x) = max_w B(x, w, l)$$

The method works quite well and fast, although it is not easy to tune all the parameters. Typically less apparent vessels emanating from strongly opacified artery are not properly detected. It is probably caused by the character of the filter and the fact that these T-junctions locally do not resemble the supposed vessel profiles. The process of validation greatly improves contrast of detected vessels but it also masks out parallel vessels, see Figure 4.6 for an example. Validation also greatly reduces non-vessel noise, see Table 4.3.

A variant of this method for bi-plane cineangiogragrams was introduced in [87].

4.2.5 Level set based methods

There are many algorithms for level set segmentation algorithms, but only a few of them were applied to vessel extraction. It seems that formulating a level set in such a way that it finds vessel-like structures is a difficult task. Let us first quickly describe what level sets are and what was the motivation to create them.

Level sets were first used by physicians to model physical phenomena of sweeping of forest fires, pouring liquids etc. [5]. In all these cases we try to follow an evolving interface. Later on level sets were recognised as a great way of segmenting objects from images. The object's boundary is represented implicitly as a *zero level set* of some function Φ:

$$\Gamma = \{ x \mid \Phi(x) = 0 \}$$

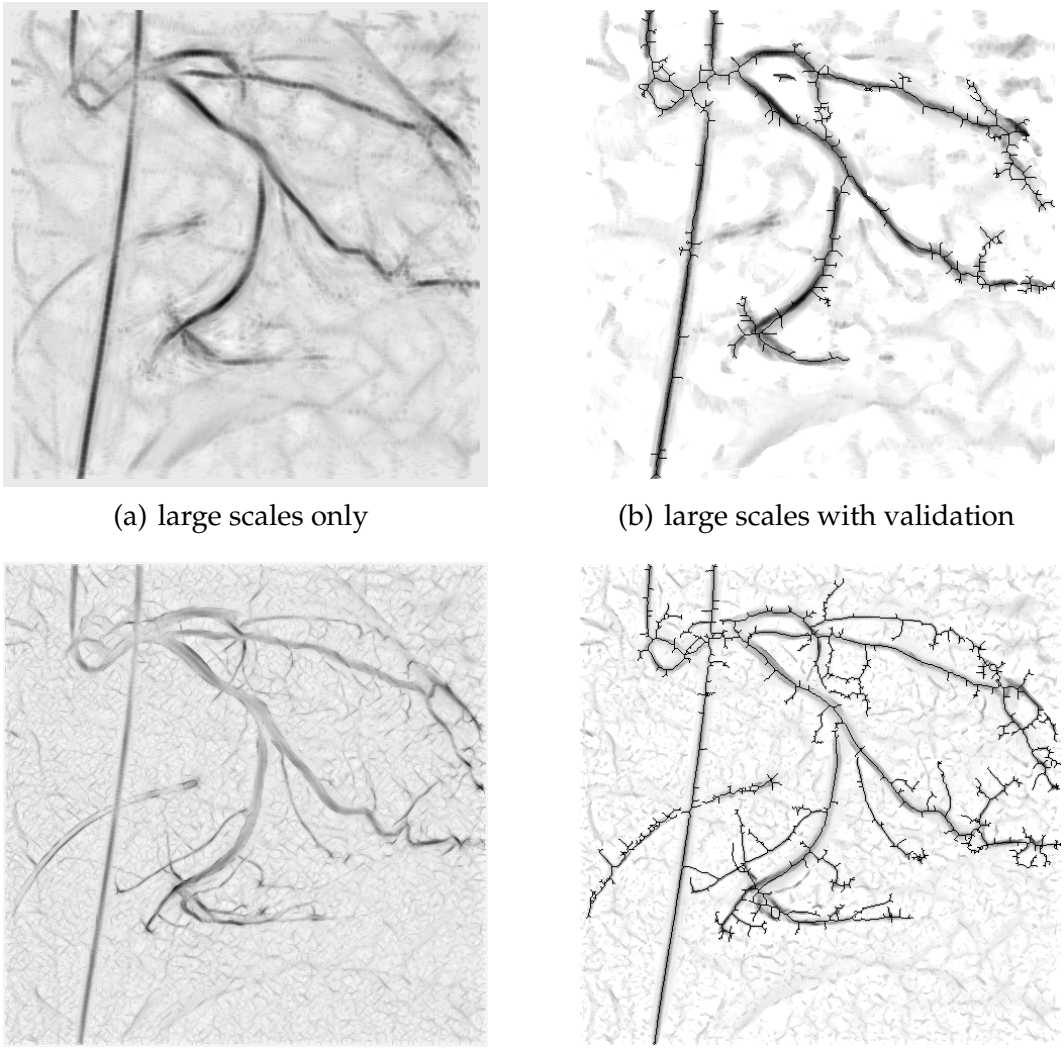
Interior and exterior of an object are then $\{ x \mid \Phi(x) \leq 0 \}$ and $\{ x \mid \Phi(x) > 0 \}$.

The evolution of an interface is controlled according to the Hamilton-Jacobi equation $\Phi_t = v|\nabla\Phi|$ to minimise some energy E. The speed that drives the segmentation is defined by a scalar field v. Often this field is defined based on the gradient of the input image because such field makes the level set stop on edges. Such v is referred to as an *edge map*. If we want to apply a shape prior, as is the case with vessel segmentation, we have to incorporate it into the velocity field v.

The main advantage of level-set segmentation over other segmentation methods and dynamic models in particular is the ability to change topology.[3] This is thanks to implicit representation, without it we would have to check for topology changes and do shape splitting and joining while explicitly constructing new parametrisations.

[2] Validation is similar to *non-maximal suppression* known from edge detectors.

[3] Note that Terzopoulos et al. improved classical snakes model to allow it for topological changes. It can be found under the names *topology adaptable snakes* or *T-snakes*. For more information on the subject see [64].



(a) large scales only (b) large scales with validation

(c) all scales (d) all scales with validation

Figure 4.6: Vessel extraction using Poli's matched filter method — vesselness along with centre-lines. If we compare the results of multi-scale results at various scales, we note right bottom corner of large scale validated enhancement (b) where the vessels are disconnected — this is caused by ringing at the same place in unvalidated image (a). For visualisation purposes the vesselness is inverted and centre-lines are shown only for validated results (b,d).

More details on fundamentals of the level set theory, concerning the Hamilton-Jacobi equation, and numeric details like up-winding scheme, signed distance function and narrow band method can be found in many articles and books [72, 73].

Recently there have been some work using level sets for vessel segmentation, capable of dealing with high level of noise and incorporating shape prior. Brieva et al. came up with a method [16] that substitutes an edge map, which is not reliable in the case of angiographic images, for an approach that models vessels and background as areas with locally Gaussian distributions with predefined mean as proposed by Chan and Vese [19]. These parameters are initialised by another method and then

recomputed in each iteration. The following functional is minimised:

$$\lambda_1 \int_\Omega \left(c_1 - I(x)\right)^2 H(x) + \lambda_2 \int_\Omega \left(c_2 - I(x)\right)^2 \left(1 - H(x)\right) + v\left(|\Gamma|\right)$$

where c_i are intensities of the two classes and v is a regularising term depending on the contour length. The authors claim precise detection of vasculature. However the reported run-times of hundreds of seconds for small images are quite prohibitive, disabling real-time usage. One more problem is the absence of any shape prior making it virtually impossible to prevent leaks.

Another interesting result is a work by Nain et al which nicely illustrates the diversity of vessel extraction algorithms [70]. The authors use not only image statistics but also a soft shape prior claiming that it greatly improves the robustness of the solution. Main problem they deal with is to restrict segmented area from leaking outside of vessels. They formulate the velocity field as:

$$v = (-\psi(x) + \kappa(x) + \alpha\epsilon_2(x, p))$$

where ψ is an image driven force, κ the curvature of Γ and

$$\epsilon_2(x, p) = \epsilon_1^p(x) + p \int_{B(x,r)} \epsilon_1^{p-1}(y)\chi(y)\ \mathrm{d}y$$

where in turn $B(x, r)$ is a ball neighbourhood and ϵ_1 is a shape prior that measures the percentage of pixels that are inside the contour out of all pixels in the neighbourhood:

$$\epsilon_1(x) = \int_{B(x,r)} \chi(y)\ \mathrm{d}y$$

for $\chi(x)$ being a characteristic function of the area inside of Γ. This effectively penalises widened areas. When the size r of neighbourhood $B(x, r)$ is selected to be approximately equal to vessel width, it helps to prevent leaks.

Although results shown in [70] are quite interesting we finally decided not to implement the method of Nain et al. mainly because there were few quite unclear things in the article that would require guessing at the right values of some parameters and their role in segmentation. These were especially calculation of the image driven force and parameter p in the description of the shape prior.

4.3 Proposed methods

Now let us describe our own variations of two common methods, a background subtraction and region growing.

4.3.1 Variants of background subtraction

Background subtraction described in Section 4.2.1 usually uses all pixels to compute background. If we know that background is bright we can model background using

only bright pixels. We take all pixels that are brighter than mean:

$$B(x,y) = I(x,y) - \frac{1}{|N'(x,y)|} \sum_{(a,b)\in N'(x,y)} I(a,b)$$

where

$$(a,b) \in N'(x,y) \text{ iff } (a,b) \in N(x,y) \text{ and } I(a,b) > \frac{1}{|N(a,b)|} \sum_{(c,d)\in N(a,b)} I(c,d)$$

Further on, we refer to this as background subtraction with a special mean.[4]

Another variation on background subtraction which we use is a filter where we enforce two criteria at once which we call naive segmentation. Two hard thresholds, one over top-hat filtered input and one over local means of input, are set. The first threshold is determined by triangle thresholding [112], the second have to be set manually.

4.3.2 Region growing

We experimented with region growing using local statistics such as:

1. local mean $M(x,y) = \frac{1}{k^2} \sum_{(a,b)\in N(x,y)} I(a,b)$
2. local standard deviation $S(x,y) = \left(\frac{1}{k^2-1} \sum_{(a,b)\in N(x,y)} \left(I(a,b) - M(x,y)\right)^2 \right)^{1/2}$
3. Canny's edge detector
4. intensity gradient $G(x,y) = |\nabla I(x,y)|$ etc.

where $N(x,y)$ is the neighbourhood. We achieved the best performance for a method with no prefiltering where the search is initialised with points that lie on boundaries of dark objects and that themselves are dark. More specifically, gradient must be above a threshold, local mean below threshold and the intensity low:

$$G(x,y) > T_g^0 \text{ and } M(x,y) < T_m^0 \text{ and } I(x,y) < T_i^0$$

From these seeds we start region growing that includes low intensity points with a gradient higher than threshold:[5]

$$G(x,y) > T_g \text{ and } I(x,y) < M(x,y)$$

By this procedure we obtain mainly the border parts of vessels, we continue by applying morphological post-processing, specifically morphological closing with small disc neighbourhood that fills in the inner part of vessels.

The method is fairly quick but gives only an approximative segmentation suitable as an initialisation for more accurate methods, see Figure 4.7. Furthermore, the method relies on several parameters that must be set manually. Working with normalised images, we use $T_g^0 = T_g = 0.02$, $T_m^0 = 0.5 \frac{1}{wh} \sum I(x,y)$, $T_i^0 = \frac{2}{3}$, neighbourhood size 31 and closing with a disc of radius 5.

[4] This special mean resembles the most an asymmetric lower 50% trimmed mean.

[5] This can be viewed as thresholding with hysteresis.

(a) reasonable segmentation (b) leaks

Figure 4.7: Vessel segmentation using our region growing method. Reasonable segmentation (a) and a result with leaks (b).

4.4 Experiments

Vessel extraction serves two purposes, to get centre-lines of thicker vessels to be able to do feature-based registration, and to be able to detect arteries to quantify arterial opacity. To meause these two qualities we designed two tests.

First, we compare the algorithm outcome V to manually segmented mask M with main arterial branches. We use four metrics, *mean-square error (MSE)*, *normalised correlation (NC)*, mean *non-vessel intensity (NVI)* and mean *non-vessel structural intensity (NVS)*. MSE is defined by:

$$\mathrm{MSE}(V, M) = \frac{1}{N} \sum (V - M)^2(x, y)$$

where we sum over pixels, N is the number of pixels, NC by:

$$\mathrm{NC}(V, M) = \frac{\sum (OM)(x, y)}{\sqrt{\sum V^2(x, y) \sum M^2(x, y)}}$$

The second two metrics are defined to be able to quantify the level of intensity for places where there should be none. Both NVI and NVS measure a false positive rate of the algorithms. NVI is defined as average intensity outside of the vessel mask:

$$NVI(V, M) = \frac{1}{|M_o|} \sum_{(x,y) \in M_o} M(x, y)V(x, y)$$

NVS uses only pixels near the edges of V and not near the edges of M:

$$NVS(V, M) = \frac{1}{|M_s|} \sum_{(x,y) \in M_s} M(x, y)V(x, y)$$

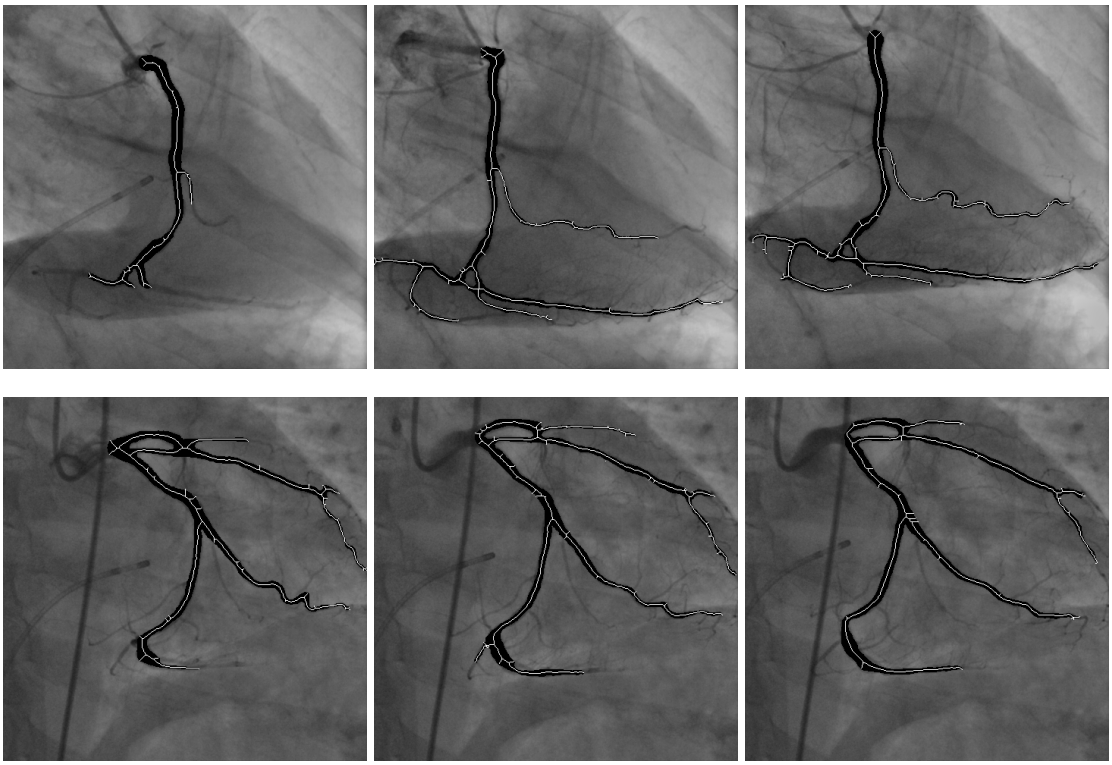


Figure 4.8: Input of vessel extraction tests. Six testing frames along with test masks (black) and skeletons (white).

where $M_o = \{\,(x,y) \mid M(x,y) = 0\,\}$ and $M_s = M_o \cap (S_V \setminus S_M)$ where S_V, S_M are dilated masks of edges in V, M.

Five test masks were created to include main arterial branches, semi-automatically by a top-hat filter with a round 19 pixel structuring element and by subsequent thresholding, and then by a manual cleanup, see Figure 4.8. So, the mask M is binary, whereas outcome V is binary or normalised depending on the algorithm.

labels	*explanation*
BG 19 P	background subtraction with P percentile (median to maximum) and 19x19 neighbourhood
BG 19 mean	background subtraction with mean
BG 19 special mean	background subtraction with special mean
Frangi $s_1, s_2, \ldots$	Frangi's algorithm with different scale settings
Poli $s_1, s_2, \ldots$	Poli's algorithm with different scale settings and with or without validation
RG segmentation	region growing method

Table 4.2: Tested vessel extraction algorithms.

Values displayed in Table 4.3 are rounded averages, labels of tested algorithms are explained in Table 4.2. Algorithms by Frangi and Poli are the best in normalised correlation. As we select only thick arteries, Poli at higher scales is better than the same at lower scales. Output of Poli is usually much more cluttered with structural noise

algorithm	*MSE*	*NC*	*NVI*	*NVS*	*time*
BG 19 100	0.033	0.478	0.094	0.100	2.918
BG 19 60	0.035	0.474	0.018	0.021	3.573
BG 19 70	0.033	0.516	0.027	0.030	3.488
BG 19 80	0.031	0.535	0.037	0.040	3.268
BG 19 90	0.030	0.533	0.051	0.056	3.125
BG 19 mean	0.036	0.523	0.011	0.015	0.303
BG 19 median	0.037	0.410	0.012	0.014	3.418
BG 19 special mean	0.030	0.608	0.029	0.033	0.728
Frangi 2,4	0.026	0.619	0.053	0.067	9.217
Frangi 2,4,6	0.026	0.616	0.062	0.073	14.420
Naive segmentation	0.125	0.543	0.125	0.322	7.058
Poli 1,2,4,8	0.036	0.421	0.096	0.097	3.163
Poli 1,2,4,8 validated	0.033	0.485	0.044	0.048	3.992
Poli 4,8	0.040	0.477	0.146	0.159	2.000
Poli 4,8 validated	0.026	0.626	0.037	0.062	2.502
RG segmentation	0.096	0.573	0.094	0.389	3.060

Table 4.3: Comparison of vessel extraction algorithms in two usual image metrics (MSE, NCC) and two metrics measuring non-vessel noise (NVI, NVS), as well as time. For explanation of NVI and NVS, see text. In each column, the best three results are underlined.

outside the supposed vessels (NVS), this gets better with validation and so the best variation of Poli is Poli 4,8 with validation, see Figure 4.6. Generally, the best results in removing non-vessels parts (NVI/NVS) are obtained for background subtraction methods. Displayed time is only indicative as our Matlab implementations are not optimised, this can be seen especially in the case of Frangi filter.

In the next test, we compare skeletons S obtained by the vessel detection methods to a manually created model M. Models were obtained by skeletonisation from masks used in the previous test, see Figure 4.8. We use three measures: *Skeleton-to-Model pixel count ratio (SMCR)*, and median/maximum of the set of *Model-to-Skeleton distances (MSD)* and *Skeleton-to-Model distances (SMD)*.

$$SMCR(M,S) = \frac{|S|}{|M|}$$

$$MSD(M,S) = \left\{ \min_{s \in S} d(m,s) \mid m \in M \right\}$$

$$SMD(M,S) = \left\{ \min_{m \in M} d(m,s) \mid s \in S \right\}$$

Thus SMCR measures how many times larger the resulting skeleton is compared to the model, median/maximum of MSD and SMD measure if all model pixels are close to any skeleton pixel and vice-versa. Results are in Table 4.4.

In the case of vesselness algorithms, we obtain the skeleton as described with Frangi's algorithm (Section 4.2.2), that is by thresholding and skeletonisation. This approach completely fails for the Poli algorithm without validation as intensity of vessels is not sufficiently higher than non-vessel intensity to be able to correctly set the thresholds, therefore we had to leave out this version of Poli algorithm.

algorithm	*SMCR*	*MSD*		*SMD*		*time*
		median	*max*	*median*	*max*	
Efstratiadis 200,0.1,0.2	7.459	1.000	39.586	35.878	221.695	313.245
Frangi 2,4	1.955	1.000	23.312	7.139	156.693	14.018
Frangi 2,4,6	2.028	1.000	29.815	9.000	146.519	19.860
Poli 1,2,4,8 validated	4.154	1.000	36.991	20.280	166.688	8.490
Poli 4,8 validated	2.362	1.443	45.903	8.198	144.135	9.170

Table 4.4: Comparison of vessel extraction algorithms in respect to skeleton extraction.

4.5 Conclusion

The results obtained in first test show that the Frangi 2,4 and Frangi 2,4,6 method and the Poli 4,8 validated method perform the best. The second test reveals a slightly better robustness of the Frangi algorithm over the Poli algorithm. Besides the automatic testing, we visually assessed the results and in our opinion the best results were provided by variants of algorithms by Frangi, Poli and our modification of background subtraction with special mean.

We conclude that the best algorithms for our purposes are the ones by Poli and Frangi, for the following reasons:

1. easy implementation within the ITK framework,
2. one of the best results in both tests,
3. visually satisfactory results in both tests,
4. multi-scale approach,
5. provides vesselness measure, and if needed it is easy to get a skeleton.

We finally selected the algorithm by Frangi over Poli's, mainly because of less non-vessel noise and better skeletons. Also, all versions of the Frangi algorithm performed well, while Poli 4,8 and Poli 4,8 with validation alternated in performance making it hard to select between the two. The Frangi algorithm was reimplemented in C++ using ITK.

Chapter 5

Registration

Image registration is one of the main areas in image processing [62] and its results are often used in medical image processing [43], as well as in other domains.

5.1 Motivation

Whenever we acquire two images of the same scene, sooner or later there will be need for integrating information from both images into one. With extensive usage of imaging methods in medicine there is often such a need: we need to follow some changes in organs given images acquired over time or after application of contrast agent [58, 40, 84], we need to compare images to atlas [27, 106], we need to merge data from different modalities to be able to display them as one [111, 18], we need to analyse motion and deformation of a tissue [76, 65], we need to reconstruct 3D objects from image slices or insert an intra-operative 2D image into 3D preoperative volume [102] etc. All of these rely on registration.

5.2 Registration overview

Often registration methods are categorised based on how correspondence is found [117], or more precisely which type of information we use to establish it. We can work directly with pixel intensities of images which is referred to as *image-based registration*, or we can extract some other type of information from the image, such as edges, corners, or contours and use it for registration which is called *feature based registration*. Here we will speak mainly about the first category.

Registration can be defined as a process of aligning two images of the same scene. It comprises two steps — finding a correspondence between a *moving image* and a *reference image* and warping the moving image to the reference image coordinates.[1]

Registration methods can be classified based on a selection of several key attributes:

- how the transformation is represented and used,

[1] Registration can be also formulated to produce the inverse transformation, but we should always prefer warping the less degraded image to maintain as much information as possible.

- how the image is warped and how to get image values at non-grid positions,
- how to evaluate the quality of the registration, and
- how to drive the process and change transform to get the best alignment possible.

Mathematically, registration is defined as follows: given two images $I_r(X_r)$ and $I_m(X_m)$ defined on $X_r \subseteq \mathbb{R}^2$ and $X_m \subseteq \mathbb{R}^2$ find a transformation T so that a functional of the form:

$$\rho\left(I_r(X_r), I_m(T(X_m), T\right)$$

is minimised. Now we will discuss the choices of T, ρ etc.

5.2.1 Transformation

Transformation representation and the transformation itself are mostly determined by the application as it can be seen as a model for the actual movement we want to recover. The simplest is a *rigid body transformation* which has the property of maintaining shapes of objects in the scene — distance between any two points does not change. It is expressed as a single rotation and single translation:

$$\vec{y} = R\vec{x} + \vec{b}$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

where R is a rotation matrix, ϕ is a rotation angle and b is a translation vector. *Affine transformation* has more degrees of freedom and allows in addition to translation and rotation also scaling and shear, and is given by:

$$y = Ax + b$$

where A is an arbitrary linear transformation and b is a translation vector. In homogeneous coordinates we can write:

$$\begin{bmatrix} \vec{y} \\ 1 \end{bmatrix} = \begin{bmatrix} A & \vec{b} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \vec{x} \\ 1 \end{bmatrix}$$

which allows for concatenating multiple affine transformations into one by multiplying the corresponding matrices in a reverse order.

To represent a deformation of a soft tissue, we have to allow non-linear transformations. These transformations maintain neither shape, collinearity nor area sizes, and are in most cases described by splines. We will speak only about *B-splines* and *thin-plate splines* as these are the ones we used.

The idea to interpolate uniformly spaced samples by a piecewise polynomial function of low order is due to Shoenberg [90]. Splines are used for interpolation because they can express complicated curves but at the same time do not suffer from

instability connected to high-order polynomials. A one dimensional *B-spline* is defined for *control points* c_i as:

$$T(x) = \sum_{i=0}^{n} B_{i,d}(x) c_i$$

where $B_{i,d}$ are basis functions of degree d recursively defined by convolution:

$$B_{i,d} = B_{i,0} * B_{i,d-1}$$

and

$$B_{i,0}(x) = \begin{cases} 1, & \text{for } x \in (t_i, t_{i+1}) \\ \frac{1}{2}, & \text{for } x = t_i \text{ or } x = t_{i+1} \\ 0, & \text{otherwise} \end{cases}$$

$B_{i,d}$ are polynomials of degree d which join seamlessly at points t_i called *knots*, so that the resulting function and its derivatives up to the order of $d-1$ are continuous. If the knots t_i are evenly spaced we speak about a uniform *knot vector* $(t_0, t_1, \ldots, t_{n+d+1})$. Further on, we consider only *uniform B-splines*, that is B-splines with a uniform knot vector. Uniform B-spline basis functions B_d are identical up to a translation, so for simplicity, we shall write $B_d(x) = B_{i,d}(x)$.

A two-dimensional transformation using B-splines of order d is expressed as:

$$T(\vec{x}) = \sum_{i=0}^{n} \sum_{j=0}^{m} \vec{c_{i,j}} B_d(x_1 - t_{i1}) B_d(x_2 - t_{i2})$$

Alternatively, B-spline can be written:

$$T(\vec{x}) = \vec{x} + \sum_{i=0}^{n} \sum_{j=0}^{m} \vec{c_{i,j}} B_d(x_1 - t_{i1}) B_d(x_2 - t_{i2})$$

which makes it easier to initialise a transform to identity, and which is the form we use. One of advantages of B-splines is that each value $T(x)$ is affected only by $d+1$ basis functions in one dimension as $B_{i,d}$ is non-zero in (t_i, t_{i+d+1}). Therefore time complexity of $T(\vec{x})$ in N-dimensions is only $\mathcal{O}(d^N)$.

Often an affine transformation is applied before a B-spline transformation because it is usually easier to recover it, then B-spline describes only non-linear part of the deformation.

Another type of splines that we use are called *thin-plate splines* (TPS), introduced by Duchon [30] and later popularised by Bookstein [15]. Inspiration comes from an analogy with a thin metal plate z supported on several places by bars of different heights whose bending energy is:

$$\iint_{\mathbb{R}^2} \left(\frac{\partial^2 z}{\partial x^2}\right)^2 + 2\left(\frac{\partial^2 z}{\partial x \partial y}\right) + \left(\frac{\partial^2 z}{\partial y^2}\right)^2$$

Mapping is defined so, that this energy is minimal while touching the bars.

In medical image processing, the positions and heights of the bars are replaced by positions of two sets of control points.[2] The transformation is given by:

$$T(\vec{x}) = \vec{a_1} + \left[\vec{a_2}^T \vec{a_3}^T\right] \vec{x} + \sum_i \vec{w_i} U\big(\|\vec{c_i} - \vec{x}\|\big)$$

where $\vec{a_i}$ model affine transformation, and $\vec{w_i}$ weigh the effect of a particular control point $\vec{c_i}$ to non-affine transformation which is modeled by *radial basis function* $U(r) = r^2 \log(r)$. Coefficients in x_1-dimension, a_{i1} and w_{i1}, are then a solution of the following linear system:

$$(c_{11}, c_{21}, c_{31}, \ldots, 0, 0, 0) = L\,(w_{11}, \ldots, w_{n1}, a_{11}, a_{21}, a_{31})^T$$

where

$$L = \begin{bmatrix} K & P \\ P^T & 0 \end{bmatrix}$$

and

$$\begin{aligned} K_{ij} &= U\big(\|\vec{c_i} - \vec{c_j}\|\big) \\ P_i &= (1, c_{i1}, c_{i2}) \end{aligned}$$

which is usually solved by QR decomposition of L [81, 109]. A limiting factor is that to find the parameters we need $\mathcal{O}(n^3)$, and to to compute $T(\vec{x})$ we still need $\mathcal{O}(n)$ with big constants, as we have to compute distances to all control points to evaluate the non-affine part of the transformation. On the other hand, one of the main advantages of TPS is that it can be easily decomposed into affine and non-affine part and that it can be easily defined by two sets of corresponding control points.

One way to define a transformation is to represent it in a form of *deformation field*. It is a vector field where for every pixel in an image there is a vector that defines its translation, so called *displacement vector*. A deformation defined in this way has the highest possible degree of freedom and can represent arbitrarily varying transformation. This has the disadvantage that we have to check the obtained deformation field corresponds to the expected transformation model. Typically, we have to include regularisation.

A transformation, in the context of image registration, are visualised either as *warped grids* or *flows*. The first representation is created by warping a regular grid, the second is a graphical version of deformation field (Figure 5.1).

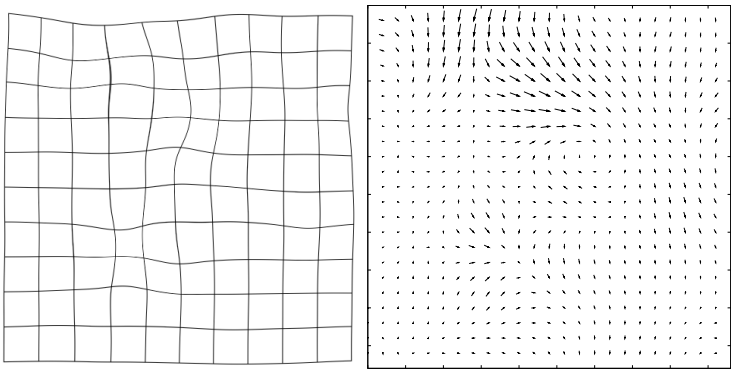
5.2.2 Metric

We need to be able to measure how precisely the moving image warped by the current transformation matches the reference image, to do so we have to define a metric comparing two images:

$$\rho\colon \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$$

When choosing a metric we have to take into account its robustness, as well as, we have to decide which differences between images we want to penalise and which are expected and correct.

[2] In the context of medical imaging, these are often referred to as *landmarks*.



(a) warped grid (b) flow

Figure 5.1: The same non-rigid transformation visualised by a warped grid (a) and a flow (b).

One of the most straightforward metrics is *mean square error (MSE)*:

$$\mathrm{MSE}(I, J) = \frac{1}{N} \sum (I - J)^2(x_1, x_2)$$

where we sum over pixels, N is the number of pixels and I, J are images. An advantage of MSE is its simplicity, on the other hand if there is a difference of multiplicative factor in intensity values MSE results in poor matches as it is based on assumption that corresponding pixels should have the same intensity. In such cases we can use *normalised correlation (NC)*:

$$\mathrm{NC}(I, J) = \frac{\sum (IJ)(x_1, x_2)}{\sqrt{\sum I^2(x_1, x_2) \sum J^2(x_1, x_2)}}$$

which does allow intensity scaling. Instead of this definition we can also take its negative value to obtain the optimum at minimum.

Also there are special metrics that can deal with images from different modalities where the relationship between intensities is not straightforward. One of these is a *mutual information (MI)* metric derived from information theory. It measures how much information the two images provide about each other. It's advantage is that we do not have to know exactly what is the functional dependence between image intensities. There are many variants of mutual information based metrics [26, 81], however the basic one is:

$$\mathrm{MI}(I, J) = H(I) + H(J) - H(I, J)$$

where

$$H(I) = -\sum p_i \log p_i$$

$$H(I, J) = -\sum p_{i,j} \log p_{i,j}$$

is an entropy, resp. joint entropy and p is a histogram-based, resp. joint histogram-based estimate of a probability density function of intensity over a given image, resp. images.

Notice that not one of these metrics takes into account spatial relations between adjacent pixels.

5.2.3 Interpolation

To be able to evaluate metric values we need to compare intensity values of moving image I_m and reference image I_r at corresponding positions. Intensity $I(\vec{x})$ is known as long as $\vec{x}$ is a grid position which is the case with I_r. However, the corresponding intensity $I_m\left(T(\vec{x})\right)$ lies at $T(\vec{x})$ which in general is not a grid position. We therefore have to extend the image function from grid positions to the whole plane $\mathbb{R}^2$, or at least some part of the plane. This is achieved by interpolation.

Commonly used interpolations include nearest neighbour interpolation, bi-linear, bi-cubic and spline interpolation [57].

Nearest neighbour interpolation is the simplest:

$$I(x) = I\left(\mathrm{round}(x)\right)$$

but the extended function I is not continuous at pixel boundaries.

Bi-linear interpolation interpolates the value given four grid values I_i at corners p_i of a cell containing x:

$$I(x) = u_1u_2I_1 + (1-u_1)u_2I_2 + u_1(1-u_2)I_3 + (1-a)(1-b)I_4$$

where u is a position of x inside the cell, in a case of unit spacing $u_i = x_i - p_{1i}$. It assumes that the intensity is linear around x and guarantees we get a continuous intensity I. Although the name may suggest it, bi-linear interpolation is not a linear operator although it is linear in both u_1 and u_2 and can be rewritten as:

$$I(x) = a_{00} + a_{01}u_1 + a_{10}u_2 + a_{11}u_1u_2$$

Linear interpolation produces results that are sometimes recognised as blurred.

Bi-cubic interpolation adds smoothness by approximating the intensity as a third order polynomial surface. Either we solve a linear system for coefficients a_{ij} where the system consists of four equations for intensity values, eight for partial derivatives and four for mixed derivatives at neighbouring nodes, or as suggested by Keys [52], we can use cubic convolution algorithm. The proposed kernel is separable, its one dimensional version is the following:

$$W(t) = \begin{cases} (a+2)t^3 - (a+3)t^2 + 1 & \text{for } 0 < |t| \leq 1 \\ at^3 - 5at^2 + 8at - 4a & \text{for } 1 < |t| \leq 2 \\ 0 & \text{for } 2 < |t| \end{cases}$$

where $W(t)$ is weight of a grid value in a distance t from x. This way we directly use 16 neighbouring grid values. Keys shows that interpolation error for $a = -\frac{1}{2}$ is $\mathcal{O}(h^3)$.

When choosing the proper interpolation method, one has to consider imposed time constraints and desired precision and smoothness of interpolation. More information can be found in a survey on interpolation methods in medical imaging [57].

5.2.4 Optimisation

The next building block needed to drive the whole process of registration is an optimiser. Transformation is gradually changed to obtain the best match (in the sense

of metric) between moving and reference images. Among the most used optimisers are gradient descent, conjugate gradient and various methods derived from the Newton's method [81, 82].

Quasi-Newton methods

Quasi-Newton methods are derived from Newton's optimisation method with the difference of only approximating the inverse of the Hessian rather than using the Hessian H directly. In some sense these methods are a generalisation of one dimensional *secant method* for root finding which approximates a derivation of f by its finite difference:

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n)$$

The update formula of Newton's method is:

$$x_{n+1} = x_n - H^{-1}(x_n) \nabla f(x_n)$$

The formula for quasi-Newton methods is almost identical with the approximation H_n^{-1} used instead of H^{-1}. The main cycle repats the four steps outlined in Algorithm 4. The quasi-Newton optimisation methods differ in the Hessian approximation update which generally has the following form:

$$H_{n+1}^{-1} = H_i^{-1} + g(H_i^{-1}, \nabla f)$$

Precise formulae are out of scope of this thesis, and can be found in [81].

Algorithm 4 Main cycle of a quasi-Newton optimisation.

Input: x_n, H_n^{-1}
Output: x_{n+1}, H_{n+1}^{-1}

obtain step s_n by solving: $H_n^{-1} s_n = -\nabla f(x_n)$
find the optimal size of step α_n by a line search in direction s_n
$x_{n+1} = x_n + \alpha_n s_n$
update the approximation H_n^{-1}

Broyden-Fletcher-Goldfarb-Shanno (BFGS) optimisation we use later in Section 5.4.2 is one of the quasi-Newton methods. It's limited-memory version is a recommend method to treat high-dimensional problems [49] commonly encountered in more complex registration tasks. The limited memory BFGS space complexity is linear compared to quadratic complexity of the original method.

5.2.5 Registration itself

A registration method is a combination of the described four building blocks. The most straightforward way is to initialise the transform to identity and run the optimiser, see

Figure 5.2. It is possible to run multiple registration methods in a chain with more and more complex transformations, eg. first recovering a rigid transformation and when it is known continuing to seek a deformation.

Another approach, referred to as *multi-resolution*, is to run the same registration methods at multiple scales, initialising each next from the results of the previous one.[3] To run at different scales input images are represented as *image pyramids*. The original images are taken and resampled to other scales, usually the scales are selected as multiples of two in all dimensions. Care must be taken when resampling, one option is to take the image, filter it with Gaussian with sigma proportional to the scale factor and resample it only then — this is referred to as *Gaussian image pyramid*. Main advantages of multi-resolution are improved robustness and speed.

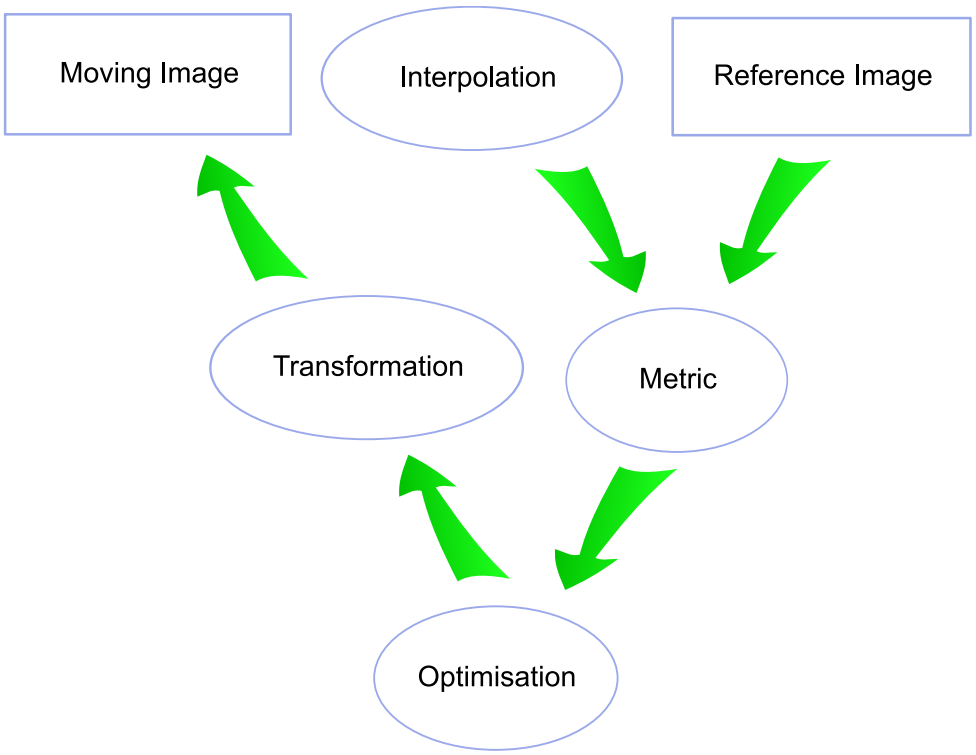


Figure 5.2: General registration method is formed by four building blocks: transformation, optimisation, metric and interpolation.

As we already stated, in many cases in medical imaging the problem requires a *non-rigid registration* which relaxes the assumptions about the transformation function, increasing the number of degrees of freedom and allowing for more complex alignments using deformable transformations. Unfortunately, this makes the problem under-constrained, so some additional knowledge must be incorporated. This is often done implicitly by choosing a basis functions of a transformation or in a form of *regularisation* which also constrains possible transformations.

[3]Sometimes this principle is described as *coarse-to-fine strategy.*

5.2.6 Movement to be compensated

One of the source of changes in our case is movement. To be able to analyse opacification we have to compensate the movement of the heart. Following is a summary of the different kinds of movement that are found in our sequences along with their geometric interpretation:

- table movement (translational)
- diaphragm movement caused by partial respiration (non-rigid, non-periodic)
- heart movement (non-rigid, approx. periodic)
- contrast agent perfusion causing an apparent motion
- movement of artificial objects

Table movement is in nearly all cases purely translational, so this can be easily compensated before the non-rigid registration step. We solve this by multi-resolution translational registration that is performed as one of the first steps after loading and preprocessing the sequence, see Section 5.4.2.

Diaphragm movement should not be present in the data at all, as patients are instructed to inspire and hold on. However, this does not hold absolutely and in some sequences we have to deal with it. We shall further discuss this issue in Chapter 4.

Heart movement should be addressed by non-rigid registration method itself, and it is the main task addressed by methods that follow in this chapter.

Contrast agent perfusion causes the vessels to ”grow” which is especially a problem for feature-based methods which mistake this growing for real stretching.

The last type of movement is also cumbersome — artificial objects such as catheter or electrode often yield strong features which are detected and used by registration. These objects can be in a position different from the position of the heart surface, and their movement does not have to correlate with the heart movement we want to compensate. Another problem of a similar type is apparent movement of a ribcage and spine in a direction opposite to movement of heart tissues. These two types of movement are connected to the data and cannot be satisfactorily solved with one plane unsychronised angiograms.

5.3 Feature-based registration

To be able to follow changes of opacity, we want to align opacified regions next to the arteries. However, these regions contain too few features which can lead the registration. Therefore, we have to assume motion in these regions is linked to the motion of arteries and if we are able to recover motion of the arteries, we will eventually know the motion of regions next to them as well. Our assumption of correlation of the respective motions is supported by a study on heart motion [60].

If we are to align the corresponding vessel-trees based on its skeletons, we need to accomplish the following steps:

1. extract vessel-tree skeleton
2. find mapping between vessel-trees of the frames to be aligned
3. extend this mapping (locally) to our region of interest

We first segment the arterial bed (or at least a significant part of it) which is discussed in detail in Chapter 4, see Section 4.2, or [71, 45, 54, 23, 83]. Once we have segmented a vessel tree (artery and its main branches), we calculate its skeleton and use it to align the frames based on the obtained contour and/or track the contour throughout the sequence. Contour correspondences found can be either used as constraints, or resampled and used directly during the alignment process [40].

If the transformation yielded by point matching is defined only for given points, we have to extend it. We can use inpainting to achieve this [8], or alternatively we can employ regularisation to get satisfactory transformation outside the skeleton, or interpolate the transformation using TPS. Motion between previous frames is sometimes used to initialise motion between subsequent frames [61]. We try to solve skeleton registration by point matching techniques which is analysed in this chapter.

5.3.1 Survey of point matching

Solving a problem of aligning two point sets closely relies on finding correspondences between points. We are given two point sets X, Y and we want to know how to map one set onto the other set. This problem is often called *point matching*.

Let us formalise it a little bit. We seek a function f for which holds:

$$f(i) = j \text{ iff point } x_i \in X \text{ corresponds to point } y_j \in Y$$

for which the energy of the following form measuring the quality of f is minimised:

$$E(f) = \arg\min_f E(T(X), f(Y))$$

Here T stands for a geometrical transformation which must be recovered, and f is extended to reorder the set Y accordingly. An example of such a functional is provided in the next section. To construct a point matching method we we have to define the energy E, and specify how to represent correspondence f and transformation T.

In medical imaging, the most frequently used method is *Iterative Closest Point (ICP)* by Besl and McKay [12] which was proposed as a solution to particular subclass of point matching where only only translation and rotation is allowed.

Iterative closest point consists of two steps which are iteratively alternated to find a plausible and satisfactory mapping. The transformation is composed of a translation t and a rotation R and a correspondence f such that:

$$E(f,(R,t)) = \sum_{x_i \in X} d\left(R(x_i) + t - y_{f(i)}\right)$$

where $d(x,y)$ is a Euclidean distance, is minimised. In the first step, we fix the transformation (R,t) and update the correspondences f, in the second step we fix the correspondence and use it to update the transformation (R,t). These steps are repeated until the transformation stops changing significantly (see Algorithm 5). If we consider only rotation and translation as a transformation, transformation definition is over-constrained for more than three points. This is solved by minimising the least-squares objective E using SVD decomposition, see [113].

Algorithm 5 ICP algorithm

Input: point sets X, Y
Output: correspondence f, rotation R, translation t

set T_0 to identity

repeat

$X' = T_k(X)$
$f(i) = \arg\min_j |x_i - y_j|$ for each point $x_i \in X'$
find T_{k+1} (R_{k+1} and t_{k+1}) using the pairs $(x_i, y_{f(i)})$

until both $\frac{||t_k - t_{k-1}||}{||t_k||}$ and $\frac{||R_k - R_{k-1}||}{||R_k||}$ are under a certain threshold (eg. 0.01)

The algorithm performs a local search, so it should be initialised close enough to the optimum to converge properly. Also note, there is implicitly an asymmetry between a template Y and an object X. If we want to introduce a symmetry we have to change the functional to:

$$E(f,g,T) = \sum_{x_i \in X} d\left(R(x_i) + t - y_{f(i)}\right) + \sum_{y_j \in Y} d\left(R(x_{g(j)}) + t - y_j\right)$$

where f, g are correspondence functions. Symmetric form is often referred to as *two-way*, and the asymmetric as *one-way*.

When can also use continuous curves as features, perhaps represented by points. In such cases, we want to use distance from a point in X to a curve defined by Y. There are two solutions — dense sampling of the curves, or approximation of tangent directions by:

$$u_{i,j} = (x_{i,j+1} - x_{i,j-1}) \,/||(x_{i,j+1} - x_{i,j-1})||$$

as proposed by Zhang [113]. Yet another way is to incorporate information about local structure via a modified distance metric as proposed by Feldmar [32] where each point is described by a $(n + d)$-dimensional vector of n differential parameters and d-dimensional position. Distance function then takes into account geometric distance and similarity at once.

Note, that naive nearest point search needs time proportional to the size of both sets, specifically $\mathcal{O}(|X||Y|)$. This can be improved by special search structures, such as *k-D trees* [28] which hierarchically divide the search space and speed-up the searches to $\mathcal{O}(|X| \log |Y|)$. Another possible speed-up to ICP methods is to subsample the point sets in the first few iterations to get a coarse registration and gradually add points as the method converges [113].

The algorithm as presented up to now neglects the fact that the point sets we obtain for the two frames usually contain *outliers*, points without a counterpart in the other image. Therefore complete matching of the two point sets is not desirable. This can be addressed by *outlier rejection* modifying the two-way distance in the following way:

$$E(f, g, T) = \sum_{x_i \in X} z_{i,f(i)} d\left(R(x_i) + t - y_{f(i)}\right) + \sum_{y_j \in Y} z_{g(j),j} d\left(R(x_{g(j)}) + t - y_j\right)$$

where $Z \in \{0, 1\}^{|X|\mathrm{x}|Y|}$ is a *correspondence matrix* specifying if x_i and y_j match or not. Then we have to simultaneously maximise:

$$\sum_{ij} z_{ij}$$

to reject trivial solution. Another way is taken by *Iterative Closest Reciprocal Point* algorithm which considers only such pairs where for the closest point $y_{f(i)}$ of x_i, the closest point in X is close to x_i as well [75]:

$$f(i) = j \text{ iff } j = \arg\min_k d(x_i, y_k) \text{ and } d(x_i, x_{i'}) < \epsilon$$

where

$$i' = \arg\min_k d(y_j, x_k)$$

In some applications, it is important to treat uncertainty in point location which is usually solved by adding a weighing factors w_i to the functional E:

$$E(f, T) = \sum_{x_i \in X} w_i d\left(R(x_i) + t - y_{f(i)}\right)$$

The weighing factor w_i is associated with a tuple $(x_i, y_{f(i)})$, hence it should be related to its uncertainty. A method to set w_i for points obtained by camera was proposed by Zhang [113].

In our case, the original ICP is not directly usable as we need to seek for a non-rigid deformation, not just a rotation and translation, we need *non-rigid ICP*. Non-rigid variants appeared long after the original ICP method, one of them was proposed

by Shelton as a means to create morphable surface models [91]. Transformation T is defined as a displacement field that maps surface X to surface Y and the the following energy functional is proposed to be minimised:

$$E(T) = E_{sim}(T) + \alpha E_{str}(T) + \beta E_{pri}(T)$$

which is a sum of similarity term E_{sim}, structure term E_{str} and prior information term E_{pri}. The ICP part of the algorithm is the first similarity term which is defined in terms of two-way distances:

$$E_{sim}(T) = \int_Z ||z - P_Y(z)|| \, \mathrm{d}z + \int_Y ||y - P_{C(X)}(y)|| \, \mathrm{d}y$$

where $Z = \{T(x) \mid x \in X\}$ and $P_A(b)$ is a point on surface A closest to point b. The other two terms of a functional serve to regularise the mapping and to incorporate prior information.

After Shelton there were other proposals for non-rigid ICP like *correlated correspondence (CC)* by Anguelov et al. [4], *Levenberg-Marquardt ICP (LM-ICP)*, or *robust point matching (RPM)* by Rangarajan and Chui [21, 20].

Levenberg-Marquardt and Iterative Closest Point Algorithm

Levenberg-Marquardt (LM) algorithm is an optimisation method to optimise a functional that takes form of sum of squares:

$$E = \big((\ldots)^2 + (\ldots)^2 + \ldots\big)$$

which can be used to express a sum of squares of distances between corresponding points $E_i = \big(d(T(x_i), y_{f(i)}\big)^2$. If we define $\vec{e}$ as a vector of residuals:

$$\vec{e}(T) = \big(d(T(\vec{x_1}), y_{\vec{f(1)}}), d(T(\vec{x_2}), y_{\vec{f(2)}}), \ldots\big)$$

and $\vec{a}$ are parameters defining the transformation T, we can write the update formula as:

$$\vec{a}_{n+1} = \vec{a}_n \; - \; \underbrace{(J^T J)^{-1} J^T \vec{e}}_{\text{Gauss-Newton}} \; - \; \underbrace{\lambda^{-1} J^T \vec{e}}_{\text{gradient descent}}$$

where J is the Jacobian matrix of $\vec{e}$ with respect to $\vec{a}$, and λ is a step taken in the gradient descent part of LM.

The described algorithm (Algorithm 6) was proposed under the name LM-ICP by Fitzgibbon [34]. The author also describes an optimised version that uses a *distance map* of reference point set to speed-up calculations of minimum distance and derivatives. Distance map can be precomputed, it is defined as:

$$D_{i,j} = \min_{y \in Y} d\left((i,j), y\right)$$

and minimum distance queries can then be evaluated in constant time.

Algorithm 6 LM-ICP algorithm

Input: point sets X, Y
Output: transformation T

initialise T to identity
initialise distance map D
while not convergence
    compute $\vec{e}$ and J
    update parameters a_{n+1} using such λ that E is reduced

Robust Point Matching

RPM like other methods interleaves correspondence matching and transformation search (Algorithm 7) [21, 20]. Authors originally proposed the method for cartoon-like contour non-rigid registration. Points are transformed using either TPS or *radial basis functions (RBF)*. Correspondence is maintained in a two-way sense, but instead of binary representation partial matches are taken into account as well. It is defined as a fuzzy correspondence matrix $Z \in [0,1]^{(|X|+1)\text{x}(|Y|+1)}$ where a cell z_{ij} defines how much the points x_i and y_j correspond to each other. All columns and rows are renormalised after each iteration[4], so that:

$$\sum_{i=0}^{|X|+1} c_{ij} = 1 \text{ and } \sum_{j=0}^{|Y|+1} c_{ij} = 1$$

The added extra column and row serve to indicate outliers. A term that ensures gradual binarisation of Z is added to the energy functional E:

$$E(Z,T) = \sum_{i=0}^{|X|}\sum_{j=0}^{|Y|} z_{i,j} d\left(T(x_i) - y_j\right) + t \sum_{i=0}^{|X|}\sum_{j=0}^{|Y|} z_{i,j} \log z_{i,j}$$

where t is the temperature of *simulated annealing* which is the chosen method for optimisation. Two more terms are added to the functional to reject trivial solutions (not all points are marked as outliers) and to regularise the transformation. TPS transformation is smoothed by Tikhonov regularisation [100] which regularises the non-affine part:

$$\begin{aligned} \vec{w} &= Q_2\Gamma \\ \vec{a} &= R^{-1}Q_1^T(Y - K\vec{w}) \end{aligned}$$

where $Q_1 \in \mathbb{R}^{|X|\text{x}3}$, $Q_2 \in \mathbb{R}^{|X|\text{x}(|X|-3)}$, $R \in \mathbb{R}^{3\text{x}3}$ is QR decomposition of X, so that $X = [Q_1 Q_2]R$, and

$$\Gamma = (Q_2^T K Q_2 + \lambda_1 I_{|X|-3})^{-1} Q_2^T Y$$

where I_k is an identity matrix of size kxk. The affine part is regularised as well yielding:

$$\vec{a} = (R^T R + \lambda_2 I_3)^{-1}(R^T Q_1^T (Y - K Q_2 \Gamma) - R^T R) + I_3$$

[4] Renormalisation is achieved using algorithm by Sinkhorn [93].

Parameters λ_1 and λ_2 are controlled by simulated annealing.

Algorithm 7 RPM algorithm

Input: point sets X, Y
Output: correspondence matrix Z, TPS transformation T

intialize T to identity, $X_t = X$, $Y_t = Y$, $t = t_{init}$
while temperature $t > t_{min}$
  calculate correspondence $z_{ij} = \frac{1}{\sqrt{t}} \exp(-||X_t - Y||/t)$
  $Y_t = ZY$
  normalise by correspondence $Y_t = \left(\ldots, \frac{y_{ti}}{\sum_j z_{ij}}, \ldots \right)$
  generate transformation T from landmarks X and Y
  use T to map X: $X_t = T(X)$
  decrease t

Apart from RPM, a non-rigid variant of ICP discussed in [21] is a variant of ICP by Feldmar which uses TPS to transform the point set [33], we refer to it as TPS-ICP. This variant of ICP includes outlier rejection in the following form:

$$x_i \text{ is outlier iff } d(x_i, y_j) > \mu + s\sigma$$

with

$$j = \arg\min_k d(x_i, y_k)$$

where μ and σ are the mean and standard deviation of all minimum distances $\{\, d(x_i, y_j) \mid j = \arg\min_k d(x_i, y_k) \,\}$. In a two-way case this is extended to consider both distances from X to Y as well as from Y to X. A similar method was presented, specifically for DSA[5] data [40].

Local structure can be also taken into account with point sets. Zheng and Doerman [114] deliver a method that is able to maintain local neighbourhood structure.

5.3.2 Experiments

We implemented three algorithms, RPM, TPS-ICP and LM-ICP described in previous section. We perform tests with several point sets. As both RPM and TPS-ICP are slow, we subsample the input point sets in the case of these two methods. The effect of subsampling on speeds can be seen in Figure 5.3. The speed-up gained by using k-D trees would pay off only with larger data sets. The main reason for high run-times is the TPS transformation, we have to keep the size of point set minimal as the time

[5] Digital subtraction angiography.

Point set	cross	skeleton	house
LM-ICP	60.4	34.2	77.0
LM-ICP with distance map	2.9	2.6	3.4

Table 5.1: For LM-ICP (run-times shown in seconds) distance map provides a significant speed-up. In comparison with TPS based methods both variants are more efficient (*cross*, *skeleton* and *house* point sets contain 407, 720 and 1247 points).

needed to solve for TPS coefficients is $\mathcal{O}(N^3)$ where N is a point set size. In our tests, TPS-ICP converges faster than RPM, and we hypothesise that RPM is not able to exploit the fact of starting close to the optimum.

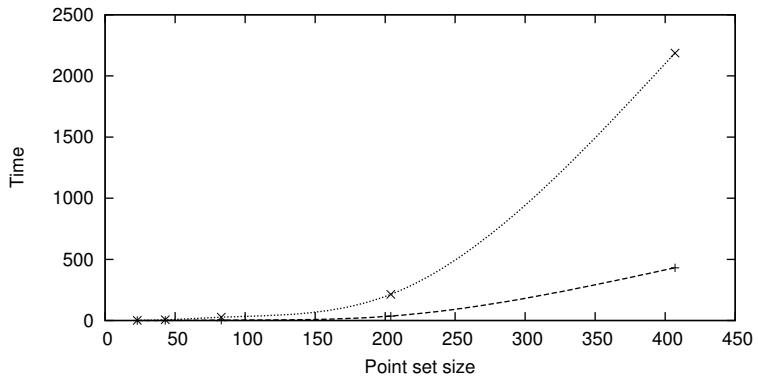


Figure 5.3: Without subsampling the input point sets both RPM (dotted) and TPS-ICP (dashed) are too slow. Only point sets with less than 50 points run reasonably fast. Results were obtained with the *cross* point set with different subsampling.

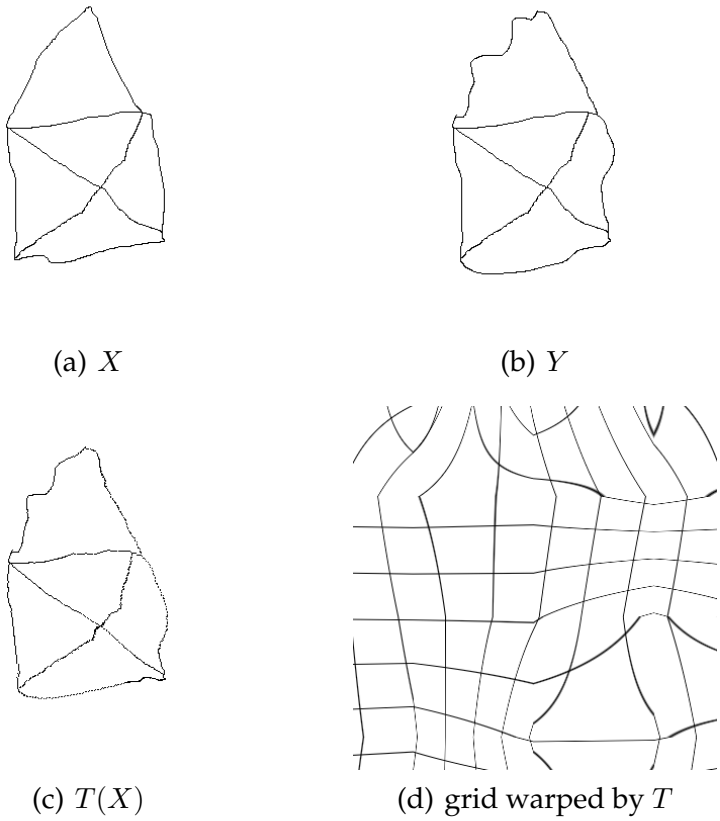
Another problem is the control of the regularising force in RPM/TPS-ICP is, either it forbids any deformation or it is too loose. Regularisation is a problem in our implementation of LM-ICP as well, we vary the order of B-splines and grid spacing but this is not enough, see Figure 5.4. Regularisation works best with RPM at the cost of longer run-times. LM-ICP is substantially quicker even without distance map optimisation, see Table 5.1.

5.3.3 Conclusion

Feature-based registration did not prove to be usable for our purposes, for the following reasons:

1. both RPM and TPS-ICP are slow to converge
2. both RPM and TPS-ICP use TPS which restricts the number of landmarks
3. the extracted vessel-tree contains vessel-like structures (such as catheter) which mislead registration

One motivation to use feature-based registration was its speed. If we were able to find the skeleton deformation and spread the information to ROI quickly we would have an



(a) X (b) Y

(c) $T(X)$ (d) grid warped by T

Figure 5.4: Results of LM-ICP with house point set, using linear B-splines with 5x5 grid restricts the transformation. We can see that the transformation represents the deformation quite well, the whole outer border of the house maps to its counterpart, only upper left border and the inner cross slightly oscillate. Explicit regularisation would probably help.

efficient method capable of on-line processing. However, we were not able to achieve this with the methods we implemented. We found it impossible to create a sufficiently smooth transformation in a given region quickly.

The main reason that finally led us to abandon feature-based approach was that we could not perform a reliable vessel extraction for all frames in a sequence. It was possible only during complete state phase which would make it necessary to generalise deformation from complete state phase to the rest of the sequence while it can be done using a deformation from a frame of correspondent heart phase in another cycle [25]. This relies on the detection of correspondent heart phase frame. Authors in the said article disposed of a synchronised electrocardiogram (ECG). We did not pursue this way as our data does not include ECG. Generally, motion estimation is studied mainly for complete state phase [61].

5.4 Image-based registration

As the feature-based approach turned out not to be feasible we started to experiment with image-based registration. There are many methods to achieve non-rigid soft tissue image-based registration. We comment just on the methods we tested and methods related to them, from the affine transformation based elastic registration [78, 77, 58] to finite element demons algorithm [99, 98, 106] or B-splines based methods [9] and [55] which turned out to yield good results in reasonable time. Finally we decided to implement a similar B-spline based method for our purposes.

5.4.1 Survey of image-based registration

Common image based registration methods follow the structure we gave in Section 5.2. In medical imaging, transformations we model are mostly non-rigid, regardless of the metric, and there is usually no close-form solution and the minimised functions are non-convex. One of the approaches to simplify optimisation and also to add regularisation is *tiling*. Image to be aligned is divided into rectangular blocks, called *tiles* and a simpler transformation is sought for the individual tiles [58, 78]. Global transform for the whole image is then reconstructed by interpolation, eg. using TPS.

Periaswamy presents such method where local geometric model is affine [77]. Apart from geometric transform, the method models intensity variations by changes in contrast and brightness.

$$a_7 I_r(x) + a_8 = I_m(T(x))$$

where

$$T(x) = \begin{bmatrix} a_1 & a_3 & a_5 \\ a_2 & a_4 & a_6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix}$$

Both models are estimated simultaneously using a standard MSE metric. First, a least-squares solution is obtained which is used to initialise a nonlinear optimisation. This is repeated at each scale, initialising the transform from the previous coarser scale. The eight parameters (six geometric and two photometric) are determined locally, and differ for each local neighbourhood. The MSE is approximated by a first-order expansion of a Taylor series. Furthermore, authors propose how to treat partially missing data using expectation-maximisation algorithm [88].[6]

Authors claim it to be universal purpose non-rigid registration method suitable for medical data [77]. As is common, it was tested with MR brain image data.

Demons registration was introduced by Thirion [99, 98],[7] originally formulated to match an object to a deformable model. In contrast to methods which are defined in some way by an attraction force only, like ICP [12], or work based on snakes [50],

[6] This extension is not important for us as the missing parts of the scene in our data are caused mainly by table shifts and can be properly treated by setting a rectangular region that is used in registration.

[7] Here demons are an analogy to Maxwell's demons from physics.

demons use both attraction and repulsion governed by optical flow:

$$v = \frac{(I_m - I_r)\nabla I_r}{|\nabla I_r|^2 + (I_m - I_r)}$$

where I_m, I_r are intensities in moving and reference image. Thirion proposes two ways how to place demons in the moving image, either in every pixel of the moving image, or only on presumed contours extracted by an edge detector. Also we have to define how to compute the forces generated by demons. First variant works with interfaces which are iso-contours and optical flow is used directly in place of forces. The registration algorithm alternates two steps, computation of forces generated by demons and update of the current transformation to reflect these forces.

As a preprocessing step, intensity values in moving image are normalised by histogram matching [49]. Histograms with a specified number of bins of both reference and moving images are matched at a number of quantile values and intensities in moving image are updated. Thresholding at mean intensity is proposed for MR data where background is mainly black, in our case this is not the case, so we use whole histograms.

We implement the method using available demons registration methods in the ITK toolkit [49] where demons are placed in every pixel. The problem is solved by a finite element method and the resulting transformation we obtain is defined as a deformation field. As with most of the methods it is important to initialise close to the optimum.

The biggest complication for our application of demons is, that the method is based on the assumption of corresponding pixels having the same intensity which is usually adjusted by preprocessing. This does not work well in our case as the corresponding intensities change throughout the image.

Next class of new non-rigid registration methods is based on level sets.[8] Bertalamio et al. developed a curve registration algorithm [11] and Vemuri et al. proposed a *level set motion registration* [106], which we shall describe here.

The authors propose to consider level sets of both moving and reference images and to let level sets of the moving image evolve in normal direction according to the difference between both images until the level sets become equal. This is achieved using an evolution equation:

$$\frac{\partial I_m}{\partial t}(x,t) = (I_r(x) - I_m(x,t))\,||\nabla I_m(x,t)||$$

This equation controls changes in intensity and at the same time another similar PDE is evolved that seeks space transformation:

$$T_t(x,t) = (I_r(x) - I_m(T(x)))\,||\nabla I_m(T(x))||$$

$I_m(x,t)$ is initialised to $I_m(x)$ and $T(x,t)$ to identity.

[8] Information about using level sets in segmentation have been already introduced in Section 4.2.5.

The method also uses preprocessing by histogram matching (see Section 5.4.1). Moreover, we found we had to add smoothing of the deformation field, and use regularisation in the form of Gaussian filtering of the deformation field. Level set motion registration as well as Demons registration are implemented in the PDE framework, where we can regularise the transform by smoothing the update field in which case we get viscous behaviour, or smoothing the deformation field directly, leading to elastic behaviour.

The method uses unsmoothed image for intensity values but gradient values are computed from a smoothed input image to increase robustness.

B-spline registration

Many non-rigid registrations are based on B-splines. B-spline transformation was shown to be amongst the best possible choices in comparison to other possible bases such as wavelets [55]. Their main advantages are effective multiresolution and implicit regularisation using grid spacing and spline order. On the other hand one of the disadvantages is the restriction of the space of allowable transformations. As for all non-rigid deformation methods, the high number of parameters makes optimisation difficult. A proper optimisation method must be chosen.

In [55], authors propose to use Levenberg-Marquardt method as an optimiser. Their elastic registration algorithm is based on B-splines not only to represent the transformation but also as a means to interpolation of the reference image. For their MR brain data, LM outperforms other optimisation methods like gradient descent (GD) or conjugate gradients (CG).

Another interesting approach is group-wise registration [68] which was recently extended to non-rigid registration using B-splines [9]. Group-wise registration is able to align a set of images simultaneously, the biggest difference is that the resulting coordinate space is determined dynamically so that no coordinate system of a specific images is preferred over the other systems. This is achieved by creating a reference image out of all the images contained in the set, usually by a pixel-wise mean.

B-spline pyramid

Multi-resolution is added to B-spline registration because of increased speed and robustness. The latter is really a concern with B-splines because the transformation in one cell of the grid is connected only with d^2 other cells through its parameters. If we start directly with an overly fine grid we may not recover some of the more global movement.

Multi-resolution approach can be applied not only to input images but to B-spline transformation representation as well [104]. As there are pyramids of images, B-spline transformations can form a pyramid as well. This way we can reduce the number of parameters in the early stages of registration and still be able to get fine registration at the end. A B-spline pyramid solves the problem how to carry over the information from one level to the next finer level in an optimal way.

When we want to get from one level to another, the first step is to extract deformation at the previous level, this gives us a vector field we want to interpolate

at next level, next we have to solve B-spline interpolation, ie. we mean recovering the parameters of a B-spline that fits the field. Unser proposes to treat B-spline coefficients as discrete signal and using digital filtering techniques [104]. Using his algorithm given coefficients c_i^l at one level we can directly compute $c_i^(l+1)$ where splines at the next level represent given transformation exactly. The most natural factor between scale of each two successive levels is an integer, eg. 2 or 2^i, where all the splines of the previous level lie in the subspace of splines of a finer level.

5.4.2 Proposed methods

Translational registration

During acquisition of angiographic sequences, the operating physician sometimes needs to adjust the view, generally this move can be supposed to be a pure translation. Therefore, prior to a non-rigid registration we seek for best rigid displacement between the frames.

This registration method uses translation as transformation, MSE, NC or MI as metric, and gradient descent (GD), conjugate gradient (CG) or Powell (PO) optimiser. Images are linearly interpolated and we use a three level pyramid of input images to make the process more robust and speed-up the convergence.

The main reason why we decided to use translational registration is that in some sequences (where there is heart rate near time sampling frequency) a result obtained by translational correction for frames of the same cardiac phase is nearly an ideal alignment so that no further non-rigid registration is needed, and even in other sequences we get closer to it.

Multi-scale B-spline registration

We use the multi-resolution B-spline registration built upon the described B-spline pyramids, Section 5.4.1 and Gaussian image pyramids, Section 5.2.5. To optimise B-spline parameters we use a limited-memory BFGS optimiser that is reported to work well in high dimensional spaces [49] whose parameters are setup at each scale separately. We use a Netlib implementation of limited-memory BFGS through ITK and vxl libraries. From our tests it is clear that limited-memory BFGS indeed needs less than a few kilobytes of memory to run while converging fast. We found out that the selection of metric does not affect the results much, therefore we use the simplest one — MSE.

A similar method that uses B-splines and LM accompanied by BFGS Hessian estimate was proposed by Sorzano et al. [95].

5.4.3 Experiments

First, we comment on the versions of translational registration. Next we study behaviour of various non-rigid registration methods with varying parameters, and at last we test non-rigid registration methods with synthetic and real data.

	MSE	*NC*	*MI*
CG	5.89*	7.89*	1.82*
GD	2.51	3.17	6.86
PO	9.67*	7.64	3.08

	MSE	*NC*	*MI*
CG	6.17*	9.19*	1.63*
GD	2.38	3.05	7.23
PO	9.35	7.59	3.15

Table 5.2: Comparison of optimisation and metric in translational registration. Displayed values are an average time (seconds) to align a frame pair based on all pairs in whole sequence. Values denoted by (*) are based upon lower number of pairs due to implementation problems, see text. Zero initialisation (left) and neighbour initialisation (right). As all the methods converge well, the method of choice is GD with either NC or MSE which is the fastest.

Translational registration tests

We tested translational registration with respect to the optimisation method and metric used. Regarding the similarity metrics (MSE, NC, MI) there were only slight differences, MI and NC work significantly better in presence of noise. However, ITK implementation of MI (MI variant by Mattes [49]) fails to produce a value when too many samples map outside the image. In optimisations, we prefer either gradient descent (GD) or Powell method (PO), where GD converges quicker but PO is more robust. Conjugate gradients method (CG) does not behave well, we think this is caused partly by numeric issues.[9] Therefore our algorithm of choice is MSE optimised by GD when we are able to initialise by neighbours, otherwise optimisation by more robust PO is preferred. The results are summarised in Table 5.2.

Furthermore, we tested the effect of initialisation. As most of the time the translation is quite small, one option is to initialise by zero. Another is to initialise by already known translations of neighbouring frames. In the second case, for a unknown translation from frame i to frame j denoted by $T(i,j)$, we use an arithmetic mean of its defined neighbours:

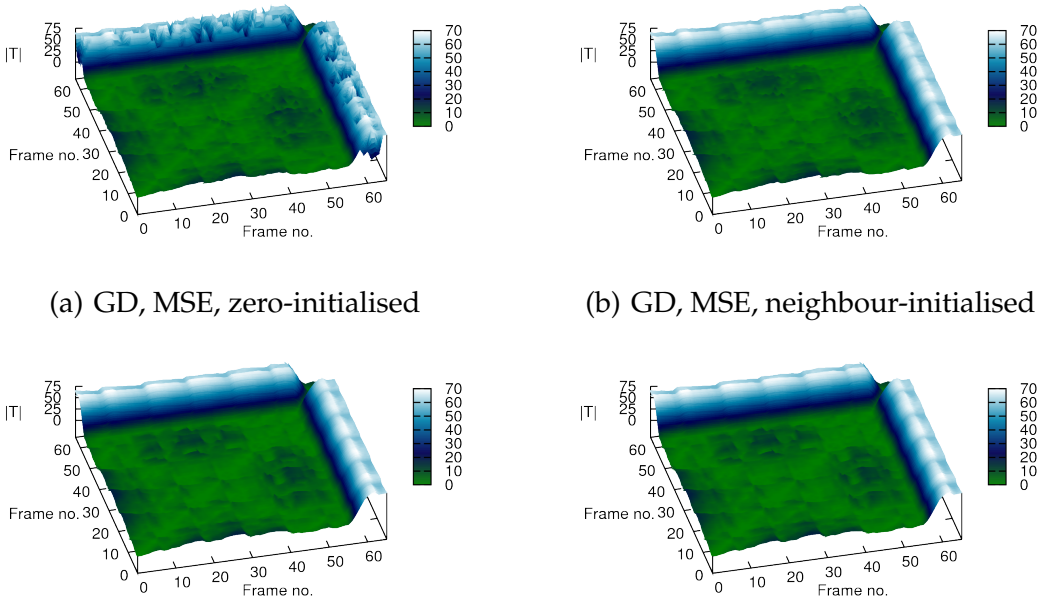
$$T_0(i,j) = \frac{1}{3}\left(T(i-1,j) + T(i,j-1) + T(i-1,j-1)\right)$$

Such initialisation makes the registration more robust compared to zero-initialisation that is prone to misalignments resulting in translation $T(i,j)$ significantly differing from its neighbours, see Figure 5.5. We think applying the translational registration is worthwhile as the time needed is low and table shifts are properly compensated, see Table 5.2. An alternative to aligning all frames is to align just one pivotting frame p to all other frames using more robust PO optimisation and then instead of $T(i,j)$ use $T(p,j) \circ T(i,p)$.

Non-rigid registration parameter tests

Next, we test various parameters of the selected methods. To compare the methods we measure run-time, MSE metric and Euclidean distance between landmarks (PTS)

[9] This is despite the fact we use Netlib implementation where these issues should be properly addressed. It seems that the implementation was not tested with noisy images, in some cases the algorithm even gets stuck in an endless loop. We were not capable of debugging the original FORTRAN code and the problem was reported to ITK developers.



(a) GD, MSE, zero-initialised

(b) GD, MSE, neighbour-initialised

(c) PO, MSE, zero-initialised

(d) PO, MSE, neighbour-initialised

Figure 5.5: Comparison of results of translational registration using zero initialisation and neighbour initialisation, see text. Figures represent a magnitude of $T(i, j)$, we can see that in the case of GD optimisation, zero initialisation causes abrupt changes (frames 55-65) whereas neighbourhood initialisation prevents them, and that it does not affect PO optimisation.

manually selected in both images:

$$PTS(X, Y, T) = \sum_i ||T(X_i) - Y_i||$$

With both PTS and MSE we also use a ratio of the metric before and after alignment: $PTS(X, Y, T)/PTS(X, Y, id)$ and $MSE(I, J(T))/MSE(I, J)$ where id is identity transform.

Tiling affine registration by Periswamy was excluded from the tests, as it is an order of magnitude slower (with run-times of 10-20 minutes) than other methods, and as such it is practically unusable. The most time consuming is the final flow interpolation that provides the smooth dense deformation. On the other hand the alignment is very good (Figure 5.6) as long as careful tuning of method parameters is done. Otherwise we end up with a distorted image instead (Figure 5.6(c)). Neither the group-wise registration is apt for our purpose as it turns out that aligning the whole sequence converges to a virtual reference frame composed of a mean of extreme systolic and diastolic frames, and aligning the other frames to this mixed frame.

Regarding our B-spline method, we experimented with various settings of parameters of the optimisation, as well as, of the B-spline transformation. There are six important parameters of the BFGS optimisation: size of the workspace, gradient convergence tolerance, search space tolerance, step length initialisation, line search accuracy and maximum number of evaluations of metric. Memory parameter defining

(a) before (b) after (c) failed

Figure 5.6: Overlay of RAO/Caudal frames (red and green) aligned by tiling affine method by Periaswamy. Before registration (a) and after successful registration (b). Alignment is nearly flawless, the only misaligned region in upper-left is not interesting for our purposes. Careful parameter tuning is essential for the result, otherwise the method fails (c).

a size of the workspace is always set to 5, and search space tolerance to 10^{-16}, as these values are recommended in Netlib source code. Gradient convergence tolerance is the most important parameter, as setting it too low does not improve results and makes BFGS optimiser take too much time. We set it to a value between 0.001 and 0.003 to balance precision versus time. Finally, the maximum number of evaluations is set to 50, as a last resort to stop the algorithm in case it converges too long, in finer scales it is further decreased as metric evaluations are more costly there.

Regarding mutliresolution, we tested the algorithm with one to four level image pyramids. More pyramid levels usually produce better output when applied to more different inputs. This sounds logical as, when there is little to no movement at the finest scale, there is even less movement at the coarser scales. In our case, the best (metric and speed) results were obtained with three level setting (Table 5.8). B-spline transformation was used generally with two levels (coarse and fine grid), save for the one level image pyramid version. The two level version uses 8x8 and 16x16 grid which means about 37 pixels per knot on the first level.

Regarding level set motion, parameters that influence registration results significantly are the number of iterations, gradient smoothing and regularisation. Level set motion is able to decrease MSE quickly and significantly but the resulting deformation field is not smooth. It aligns everything including parts that should not be aligned. This especially happens when in one frame there is an opacified artery whereas in the other one it is not. It can be partially improved by applying regularisation at each step of registration which increases the computational time and sometimes causes oversmoothing. Unfortunately, PTS does not get better with more regularisation, see Figure 5.7. The optimal value of gradient smoothing is $2 \leq \sigma_g \leq 4$, see Table 5.3.

Regarding demons registrations, parameters that influence registration results sig-

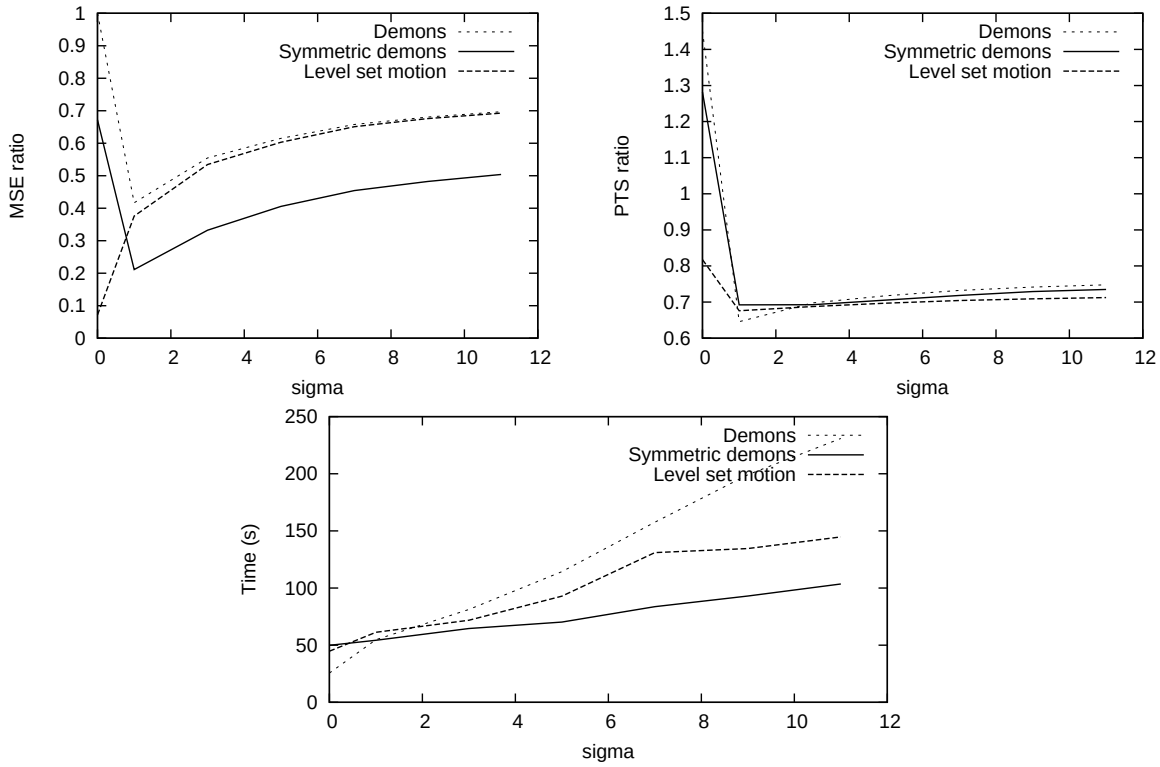


Figure 5.7: Effect of regularisation in level set motion and demons registration.

σ	0	2	4	6	8	10
PTS	72 ± 21	79 ± 25	82 ± 18	84 ± 8.5	79 ± 12	79 ± 15
MSE	84 ± 8.4	8.4 ± 2.1	7.2 ± 2.0	8.3 ± 1.7	11 ± 2.1	12 ± 3.0

Table 5.3: Level set motion gradient smoothing should use $2 \leq \sigma \leq 4$. Displayed values are ratios of MSE/PTS before and after alignment in percents obtained for various settings of σ.

nificantly are the number of iterations, symmetry of forces. The effect of regularisation in either demons registration, or symmetric demons registration is weak as in level set motion registration. Setting regularisation $\sigma > 1$ does not further improve the results. On the other hand run-time does not increase as quick as with level set motion. The only usable value is smoothing with $\sigma = 1$.

Preprocessing by histogram matching in level set motion and demons registration mostly does not influence the results. There is nearly no change in intensity between results of preprocessing with different numbers of histogram levels, see Figure 5.8 and the differences in alignment due to different preprocessing is also negligible (Table 5.4). Runtimes are not influenced by any parameter setting of preprocessing. It does not seem worthwhile to setup number of histogram levels to a number higher than the number of input intensity levels, as anything more gains only from the intensity levels created in interpolation during resampling. In the case of match points optimal count

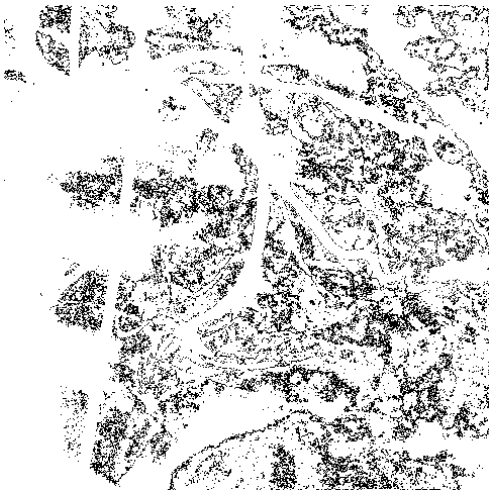


Figure 5.8: Difference of preprocessing results in level set motion with 64 and 512 histogram levels is small. Highest difference is 1 in 255, image was rescaled for visualisation.

is approximately 8. Note that the differences in run-times are very small and can be attributed to cache performance or similar issues.

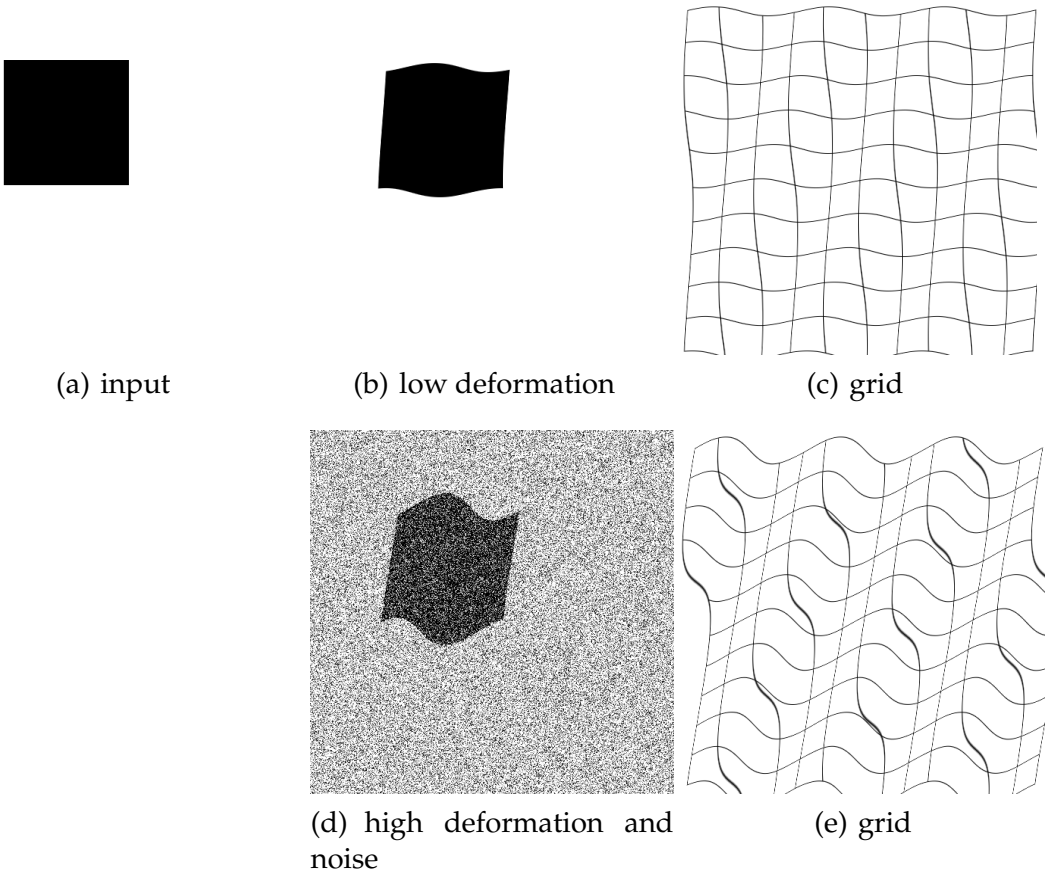
histogram levels	*64*	*128*	*256*	*512*	*1024*
Level-set motion	6.9 ± 1.9	7.0 ± 1.9	7.1 ± 1.9	7.2 ± 1.9	7.2 ± 1.9
Demons	42 ± 12	42 ± 12	42 ± 12	42 ± 12	42 ± 12
Symmetric demons	21 ± 5.3	21 ± 5.3	21 ± 5.3	21 ± 5.3	21 ± 5.3

Table 5.4: Parameters of preprocessing in PDE deformable registrations do not influence results significantly. Displayed values are mean and standard deviation of MSE ratios between and after alignment in percents. A little better results for lower number of histogram levels is probably caused by rounding errors (in the case of Demons algorithm, changes were even smaller). Analogical table without important differences was obtained for the changing number of match points.

Non-rigid registration comparison

The input to the first comparison test was a simple artificial image translated and deformed (Figure 5.9). The second comparison test uses the same synthetic deformations applied to a real angiographic frame (Figure 5.10(e)). In both tests we also studied the performance of the methods when noise was added. Results were compared by accuracy defined as a Euclidean distance between known deformation field and the recovered field. In the third test, we align different angiographic frames, these are selected so that they cover the range between opacified and unopacified frames, frames from approximately the same cardiac phase (small deformation) and for different cardiac phases (large deformation) (Figure 5.10). In all three tests non-rigid registration method was initialised with translation recovered by our translational registration.

In the Test 1 (Table 5.5), demons and level set motion registration achieve the best results. In the case of level set motion more time is needed with increasing



(a) input (b) low deformation (c) grid

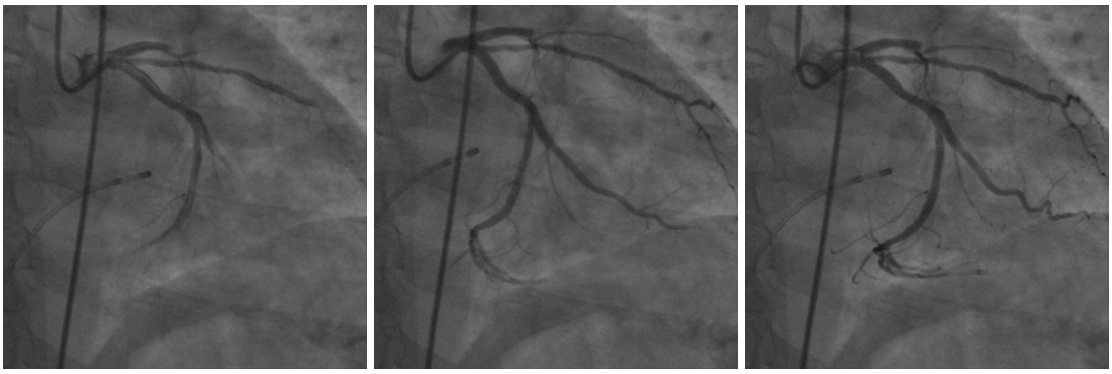
(d) high deformation and noise (e) grid

Figure 5.9: Test 1 — input.

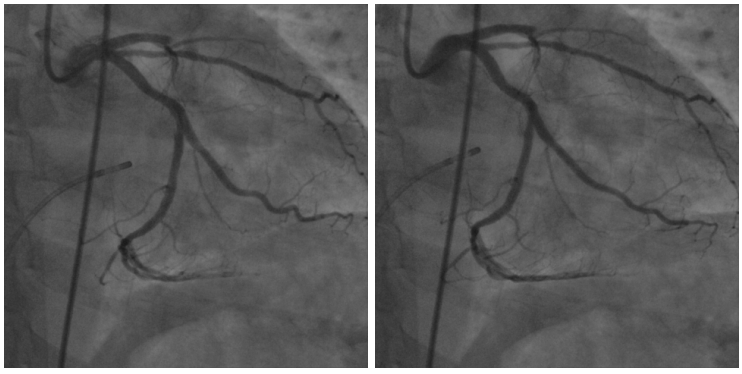
regularisation. Level set motion with no regularisation is the quickest but the resulting deformation is unsmooth. B-spline registration runs quicker than other methods but its accuracy is approximately twice smaller in the case of non-rigid deformations. In case of translations and identity, all algorithms take time to recognise there is nothing left to align. In this case B-spline methods are more accurate than other methods unless too much noise is added, in which case especially the four level B-spline method fails. This is probably caused by tolerance parameter of the L-BFGS optimisation.

Compared to the Test 1, in the Test 2 we can see a significant improvement of the B-spline method in the case of deformations where accuracy of all methods is approximately the same, Table 5.6. In the case of only translations and translations with noise B-spline method is many times better, as is the case of run-time where our B-spline methods are the quickest. Three level B-spline seems to be the best choice in the matter of speed and metrics.

The results of the Test 3 (Tables 5.7 and 5.8) show that B-splines method is a little worse at aligning frames from the same cardiac phase but in contrary to other methods it also provides the best registration in case of frames which differ significantly. Note that even the best alignment obtained for such frames reduces the movement of the selected landmarks as measured by PTS to about 50%. On the other hand, such motion



(a) frame 11 (b) frame 16 (c) frame 22



(d) frame 23 (e) frame 27

Figure 5.10: Test 3 — input frames from different angiographic and cardiac phases, we use (a)-(c), (a)-(d), (b)-(d), (b)-(e).

compensation for frames of the same cardiac phase is reached by all methods. We can see that symmetric Demons iteration is approximately four times longer than its asymmetric version. Both Demons methods converge slow for distant frames.

Multiresolutional B-spline method, with a properly tuned BFGS optimiser, seems to be the best choice for our purposes as it produces smooth mapping with consistent minimisation of both MSE and PTS in 10–30 seconds. Although regularisation based on grid refinement is not that flexible, it is effective in both speed and quality, see Figure 5.11.

5.4.4 Conclusion

We tested the registration methods described in Section 5.4.1, found best values of parameters for each of the methods and compared the results. We remark that all the methods provide incorrect or insufficient alignment when there is too much structural change, either in the position of the arterial tree or in case where the electrode moves differently than the rest of the scene producing a deformation that is not smooth. Also displacements of vessels during ventricular systole are really large (20–30pixels, approx. 4–6% of the field of view), so that even between consecutive frames the differences are too large. Therefore, we try to avoid the need for such alignment, and

	i		*t*		*t, n=0.5*		*t, n=1*	
	accuracy	*time*	*accuracy*	*time*	*accuracy*	*time*	*accuracy*	*time*
B-spline 1 level	0.031	34	0.012	15	0.047	50	0.044	57
B-spline 2 levels	0.000	41	0.006	10	0.027	53	0.095	47
B-spline 3 levels	0.000	46	0.008	14	0.043	44	0.842	48
B-spline 4 levels	0.000	46	0.009	22	4.108	23	4.135	45
Demons	0.030	78	0.013	75	0.044	79	0.026	78
Symmetric demons	0.217	59	0.092	57	0.380	60	0.402	61
*Level set motion**	0.972	30	0.014	1	1.242	42	1.420	47
*Level set motion**	0.619	47	0.013	17	0.710	69	1.108	72
*Level set motion**	0.375	58	0.013	21	0.458	97	0.639	86

	t, d		*t, d, n=0.5*		*t, d, n=0.5*	
	accuracy	*time*	*accuracy*	*time*	*accuracy*	*time*
B-spline 1 level	11.582	82	11.195	82	11.542	61
B-spline 2 levels	35.205	34	35.145	48	24.833	67
B-spline 3 levels	21.835	48	19.493	40	19.700	49
B-spline 4 levels	18.843	47	21.488	45	32.893	52
Demons	11.245	78	11.344	79	11.456	81
Symmetric demons	11.219	60	11.341	62	11.467	62
*Level set motion**	11.260	24	11.510	41	11.664	42
*Level set motion**	11.086	67	11.208	69	11.435	72
*Level set motion**	11.105	82	11.215	85	11.396	88

Table 5.5: Results of the Test 1. First table shows how algorithms recover translation, *i* stands for identity, *t* for translations, *n* is the amplitude of added white noise. Second table shows how algorithms recover deformation (*d* stands for deformation). The time is the time elapsed for non-rigid registration (excl. translational registration). (*) level set motion algorithm with increasing regularization, $\sigma \in \{0, 3, 5\}$.

use reasonably similar input images selected by a technique we call matching frames (Section 6.2.1).

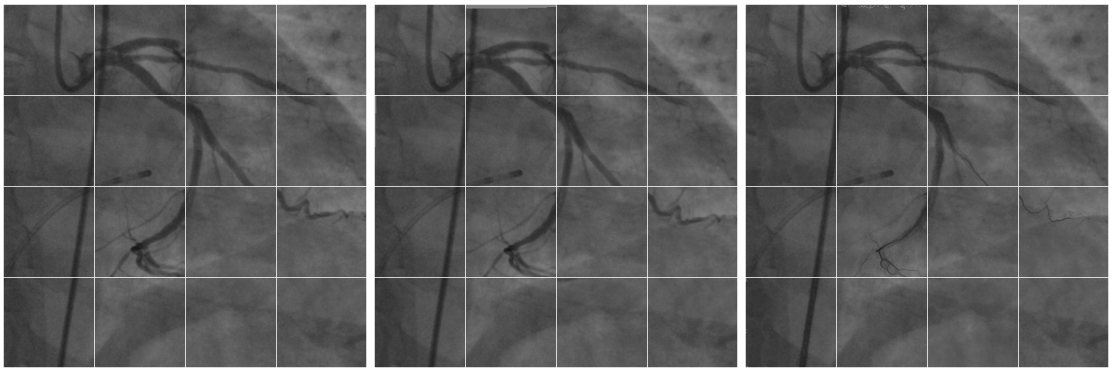
Alignment of similar frames pertaining to the same cardiac phase is much more successful. From the comparison of the methods, we conclude that three level B-spline registration is the best for our purposes and we will use this method further on.

One possible improvement of the registration could be to take into account that some objects (such as bones) should behave rigidly, eg. by adding some rigidity constraint [86].

	i		*t*		*t, n=0.1*	
	accuracy	*time*	*accuracy*	*time*	*accuracy*	*time*
B-spline 1 level	0.022	22.446	0.090	37.637	0.660	51.635
B-spline 2 levels	0.000	24.186	0.111	18.044	0.567	46.552
B-spline 3 levels	0.000	4.795	0.108	19.022	0.714	45.812
B-spline 4 levels	0.000	1.647	0.210	24.078	0.649	55.559
Demons	0.289	78.811	1.195	79.200	0.849	80.058
Symmetric demons	0.457	59.165	1.652	60.106	1.137	59.759
*Level set motion**	0.789	21.043	3.442	33.164	2.395	41.718
*Level set motion**	0.552	36.967	1.511	51.521	1.570	71.400
*Level set motion**	0.361	49.463	1.051	64.308	1.107	89.870

	t, d		*t, d, n=0.1*	
	accuracy	*time*	*accuracy*	*time*
B-spline 1 level	6.276	38.020	5.184	33.576
B-spline 2 levels	6.567	41.301	6.224	31.059
B-spline 3 levels	6.505	34.789	5.969	16.408
B-spline 4 levels	6.196	41.918	6.795	53.726
Demons	5.830	80.309	6.775	79.869
Symmetric demons	6.172	59.958	6.872	60.207
*Level set motion**	8.074	41.519	7.749	40.832
*Level set motion**	6.133	69.557	6.373	68.154
*Level set motion**	6.126	86.421	6.361	84.946

Table 5.6: Results of the Test 2. First table shows how algorithms recover translation, *i* stands for identity, *t* for translations, *n* is amplitude of added white noise. Second table shows how algorithms recover deformation (*d* stands for deformation). The time is the elapsed time for non-rigid registration (excl. translational registration). (*) level set motion algorithm with increasing regularization, $\sigma \in \{0, 3, 5\}$.



(a) before alignment (b) B-spline (3) (c) symmetric demons

Figure 5.11: Checkerboards of non-rigid registration results of frames 11 and 22. We can see the importance of regularisation — B-spline method properly leaves opacified regions of arteries undeformed where there is no counterpart in the early frame, demons registration with insufficient regularisation squishes these parts of vessels to obtain lower metric values. We get similar artifacts with level set motion unless we regularise intensively.

	Frames 16x23			*Frames 11x22*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.8e-3	22.9(7.7)		1.0e-3	5.14(1.94)	
Demons (25)	1.3e-3	19.3(7.3)	6.6	4.9e-4	1.86(1.85)	6.8
Demons (50)	1.1e-3	19.2(7.6)	13	4.5e-4	1.72(1.59)	13
Demons (100)	9.0e-4	19.3(7.4)	26	4.2e-4	1.60(1.44)	26
Demons (200)	7.9e-4	19.5(7.3)	52	4.0e-4	1.50(1.14)	54
Demons (400)	6.9e-4	19.7(7.5)	107	3.9e-4	1.38(0.91)	104

	Frames 11x23			*Frames 16x27*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.6e-3	20.1(6.4)		6.9e-4	4.42(1.32)	
Demons (25)	1.2e-3	18.2(6.9)	6.8	2.5e-4	2.32(1.48)	6.9
Demons (50)	1.1e-3	18.2(6.9)	14	2.2e-4	2.47(1.63)	14
Demons (100)	9.3e-4	18.1(6.9)	26	2.0e-4	2.51(1.60)	30
Demons (200)	8.6e-4	17.5(7.1)	53	1.9e-4	2.52(1.66)	61
Demons (400)	8.4e-4	17.3(7.6)	104	1.9e-4	2.50(1.68)	119

	Frames 16x23			*Frames 11x22*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.8e-3	22.9(7.7)		1.0e-3	5.14(1.94)	
Symmetric demons (25)	4.8e-4	19.2(7.0)	27	2.2e-4	2.35(2.55)	27
Symmetric demons (50)	4.0e-4	19.3(6.9)	54	2.1e-4	1.86(1.17)	54
Symmetric demons (100)	3.6e-4	19.4(6.9)	107	2.0e-4	1.57(0.71)	107
Symmetric demons (200)	3.3e-4	18.3(7.1)	215	2.0e-4	1.59(0.71)	224
Symmetric demons (400)	3.2e-4	18.2(7.2)	453	2.0e-4	1.60(0.72)	460

	Frames 11x23			*Frames 16x27*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.6e-3	20.1(6.4)		6.9e-4	4.42(1.32)	
Symmetric demons (25)	4.7e-4	18.3(7.2)	27	1.1e-4	2.66(1.71)	27
Symmetric demons (50)	4.1e-4	18.3(7.4)	54	9.7e-5	2.92(1.62)	56
Symmetric demons (100)	3.8e-4	18.6(8.5)	108	9.3e-5	2.91(1.68)	110
Symmetric demons (200)	3.7e-4	18.6(9.3)	229	9.2e-5	2.90(1.67)	229
Symmetric demons (400)	3.6e-4	17.8(9.0)	453	9.1e-5	2.89(1.68)	463

	Frames 16x23			*Frames 11x22*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.8e-3	22.9(7.7)		1.0e-3	5.14(1.94)	
Level set motion (25)	9.4e-4	19.373(7.0759)	14	4.3e-4	2.40(2.33)	15
Level set motion (50)	7.9e-4	19.185(7.0584)	29	4.0e-4	2.34(2.28)	28
Level set motion (100)	6.6e-4	18.926(7.1545)	68	3.7e-4	2.24(2.13)	62
Level set motion (200)	5.5e-4	18.687(7.3571)	114	3.5e-4	2.15(2.01)	119
Level set motion (400)	4.7e-4	18.34(7.5076)	227	3.5e-4	2.22(1.93)	226

	Frames 11x23			*Frames 16x27*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.6e-3	20.1(6.4)		6.9e-4	4.42(1.32)	
Level set motion (25)	7.9e-4	18.2(7.6)	14	2.8e-4	2.43(1.35)	14
Level set motion (50)	7.0e-4	18.0(7.8)	28	2.5e-4	2.33(1.37)	29
Level set motion (100)	6.4e-4	18.1(8.1)	60	2.4e-4	2.41(1.31)	55
Level set motion (200)	5.6e-4	18.4(9.2)	123	2.3e-4	2.60(1.31)	113
Level set motion (400)	5.5e-4	18.7(11.0)	225	2.3e-4	2.72(1.40)	228

Table 5.7: Test 3 - real angiogram data. Measured values are MSE, PTS (avg&std) and time in seconds. Both demons algorithms use regularization $\sigma = 1$, whereas level set motion was run without regularization. Data in the left column shows frames from different cardiac phases, in the right column from approximately the same phase.

	Frames 16x23			*Frames 11x22*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.8e-3	22.9(7.7)		1.0e-3	5.14(1.94)	
B-splines (1)	8.8e-4	17.9(7.1)	31	5.4e-4	2.99(2.52)	20
B-splines (2)	8.6e-4	11.9(5.1)	32	5.5e-4	2.82(2.52)	22
B-splines (3)	8.5e-4	15.2(5.9)	13	5.5e-4	2.66(2.26)	23
B-splines (4)	7.9e-4	14.7(5.6)	63	5.1e-4	2.70(2.34)	38

	Frames 11x23			*Frames 16x27*		
	MSE	*PTS*	*time*	*MSE*	*PTS*	*time*
reference	1.6e-3	20.1(6.4)		6.9e-4	4.42(1.32)	
B-splines (1)	9.7e-4	19.0(11.0)	21	3.5e-4	2.48(1.72)	18
B-splines (2)	9.9e-4	14.5(9.3)	23	3.5e-4	2.37(1.65)	29
B-splines (3)	9.8e-4	14.3(9.9)	19	3.5e-4	2.44(1.70)	26
B-splines (4)	9.2e-4	8.07(4.30)	33	3.4e-4	2.75(1.60)	39

Table 5.8: Results of the B-spline method with varying number of pyramid levels. Measured values are MSE, PTS (avg&std) and time in seconds. Data in the left column shows frames from different cardiac phases, in the right column from approximately the same phase.

Chapter 6

Opacity quantification

Problem of semi-automatic to or fully automatic opacity quantification in X-ray angiographic data has been studied scarcely so far. One of the reasons is that opacity quantification in angiographic imaging started evolving rather late, compared to angiography itself, during nineties [89], TMP scale itself was defined in 1999 [38].

6.1 Prior-art in angiographic image processing

To our surprise we have found very few articles dealing with opacity quantification, ie. properly aligning frames in an X-ray angiogram sequence, defining region of interest and analysing the change of opacity in that region.

Most of the work concerning X-ray angiography and angiography image processing in general are targeted at vessel enhancement or vessel extraction, as presented in Chapter 4. Some interesting work have been done in order to reconstruct 3D arterial tree from angiographic data. Common concept is to use bi-plane angiographic imaging and epipolar geometry [71, 69]. Reconstruction from single plane imaging is difficult and it was studied scarcely, Messenger et al. developed a method that uses several single-plane views carefully selected by a skilled physician so that there is minimal inter-vessel shadowing [66]. Authors claim to be able to reconstruct 3D arterial tree.[1] Preprocessing preceding reconstruction is often some form of coronary tree labelling like in [45].

Reconstructed arterial tree is usually used in an area called *quantitative (coronary) angiography (QA)*.[2] It includes quantification of anatomic features, among other estimation of vessel width, bifurcation angle or radius of curvature etc. [94]. Especially in coronary angiography, this is done in many cases for automatic stenosis detection, or to follow atherosclerotic changes [48]. Cerebral arterial tree, fluoroscopic images of retinal arterioles and venules [13], or pulmonary and brachial angiographic images are analysed as well. Another application is a fusion of pre-interventional cine-angiograms, where the arterial tree is visible, with interventional fluoroscopic data in order to be able to improve catheter guidance during the percutaneous transluminal

[1] It is not clear whether they use CT, MR or X-ray angiography.

[2] QA should not be confused with opacity quantification.

coronary angioplasty (PTCA) [22]. Similarly the fusion of brain angiographic images with 3D X-ray rotational angiographic volumes was also studied [51, 6]. Another usual area of investigation is cardiac deformation analysis, some information can be found in a review by Meijering et al. [65]. Apart from angiography, there is a wealth of literature studying movement of the heart heart and especially segmentation of the left ventricle in other modalities.

The only prior art we could find that is of greater relevance for our application is a work by Condurache et al. [25] which deals with tracking of a region of interest (ROI) in angiogram sequence. In the method, feature points in vessels are detected first, then these points are tracked throughout the sequence and displacements vectors generated by their movement drive the actual ROI deformation.

Feature points are defined as locations where the two local orientations are prominent. These are detected by thresholding an image obtained as the small eigenvalue λ_2 of the *single orientation tensor* [3]:

$$T_1 = \int_\Omega \nabla I \left(\nabla I\right)^T \, \mathrm{d}\Omega = \int_\Omega \begin{pmatrix} I_x^2 & I_{xy} \\ I_{yx} & I_y^2 \end{pmatrix} \, \mathrm{d}\Omega$$

where Ω is local neighbourhood. Threshold is set by the Otsu method [74] and from the thresholded image feature points are selected by non-maximal suppression. To track them throughout the sequence a similarity is defined as:

$$s = w_1 s_p + w_2 r + w_3 s_o$$

where w_i are weights, s_p is the Euclidean distance between feature points, r the correlation coefficient of their local neighbourhoods and s_o is a similarity of their local orientations obtained as *mixed orientation vector*:

$$\vec{a} = \left(\cos(\phi_1), \cos(\phi_2), \sin(\phi_1 + \phi_2), \sin\phi_1 \sin\phi_2\right)^T$$

where ϕ_i designate two local orientations. Vector $\vec{a}$ is obtained as an eigenvector corresponding to the lowest eigenvalue of the *multiple orientation tensor* [3]:

$$T_2 = \int_\Omega \begin{pmatrix} I_{xx}^2 & I_{xx}I_{xy} & I_{xx}I_{yy} \\ I_{xy}I_{xx} & I_{xy}^2 & I_{xy}I_{yy} \\ I_{yy}I_{xx} & I_{yy}I_{xy} & I_{yy}^2 \end{pmatrix} \, \mathrm{d}\Omega$$

Feature points from two consecutive frames are considered a true pair only if they have mutually the highest similarity among all other candidates, the unsuccessful pairs are considered outliers and using a significance test a threshold on similarity to select the finally used pairs is determined.

Then the ROI, input by user, is deformed according to the displacement of the selected feature points using an energy minimisation method similar to snakes [50]. Energy is defined as:

$$E = E_{ext} + E_{el} + E_s$$

where internal energy $E_{el}+E_s$ is modelled as elasticity and stiffness in snakes. External energy

$$E_{ext} = \sum_i \sum_j d_{ij}^2$$

is summed over all points at ROI boundary and all feature points. Each boundary point is connected to each feature point by a string whose length is initialised at the beginning of optimisation. The string prolongation d_{ij} is set to the displacement of the feature point from its position in the previous frame.

The method is used to track the feature points only during a complete state phase and the results are generalised to the rest of the sequence, ROI transformation of the corresponding heart phase is chosen using synchronised ECG data. Authors base this usage of ECG upon previous research of mask selection in DSA [97].

The authors admit that ROI tracking is hard because of lack of good features and that their method does not yield large overlap between subsequent ROIs. They also find out that ROI is translated according to features found but shape deformation of ROI is not achieved. They claim that this is not a serious problem because the neighbouring areas are often comparable in opacity as they are covered by capillaries stemming from the same origin. Concerning evaluation of opacity, the article [25] is rather concise as the main subject of the article is ROI tracking. It seems the authors propose to evaluate opacity by visual inspection of ROI histogram evolution throughout a sequence. They also propose to use a low quartile of the histogram to do such inspection.

Our ROI tracking method is different to that of Condurache et al. in the following aspects. We solve ROI tracking in another way by directly aligning the images. This is better in the fact that deformation of ROI can accurately account for the non-rigid deformation of the tissue. We also propose multiple ways on choosing proper masks that are used to improve readability of opacity curve which can be considered a variation of DSA. Further, we have to deal with missing synchronised ECG data which makes it difficult to compare our results to their algorithm as input data is different.

Condurache et al. developed two more interesting complementary methods: to detect the respiration status [24] and to identify complete state frames [2]. In the first method respiratory status is considered to be linked with diaphragm which is located by a Hough transform applied to a smoothed edge map assuming that the diaphragm is approximately circular. Authors consider some anatomical constraints about size and position of diaphragm in given projections. The only problem of the method is that it relies on diaphragm being visible in a sequence which is not true for all the projections. An alternative method to detect breathing could be adapted from our cycle detection, discussed further in this chapter, Section 6.2.1.

Second method serves to identify frames with an opacified arterial tree [2]. All frames are first preprocessed by a vesselness filter, similar to background subtraction, Section 4.2.1. Supposing all preprocessed images contain two classes, a curve consisting of 95-percentile of intensity is extracted. The obtained feature curve is then segmented, for which purpose the authors propose two methods, they either

fit three polynomials using *maximum-likelihood criterion (ML)*, or model the curve as a *hidden Markov model* and estimate the transition between the two classes by *maximum a-posteriori*-criterion.

6.2 Proposed methods

We propose two methods — opacity enhancement which is similar to DSA, and opacity curve extraction and assessment. Enhancing opacity is achieved as follows: frames in the sequence are aligned as described earlier in Chapter 5, accounting for rigid as well as non-rigid deformation, and then a background is subtracted. Opacity curves extraction needs vessel detection as described in Chapter 4 to estimate the vesselness and perfusion of arteries which serve as a reference to measured opacity of the selected region of interest.

6.2.1 Opacity enhancement

One of the key ideas in our method is to complete align all images effectively to cancel the motion and to work with such "suspended" sequence. In case of perfect alignment, the task would be close to a usual DSA where there is a frame without contrast agent which is subtracted from frames with contrast agent. Unfortunately, good and reliable image alignment of all frames that, aligns the arteries, and at the same time maintains subtle details carrying information about opacification, is difficult as shown in evaluation of registration methods (Section 5.4.3). Another complication to be considered is that in the first few frames there can be some opacification resulting from previous injection. These findings lead us to the idea that registration does not have to be performed between all frames, at least for our purposes.

Matching frames

A correct selection of reference frames is crucial for any enhancement. As we have shown in Section 5.4.3 the alignment of frames pertaining to the same cardiac cycle phase tends to be more successful, as the corresponding frames differ less in the position of arteries. To use the best alignments available we should prefer these frames. At the same time we can restrict the alignment to these frames avoiding needless computation.

For each frame, we select frames with highest normalised correlation and search for these frames so that they are approximately equidistant along the sequence, see Figure 6.1. We refer to these frames as *matching frames* and formally define n-th matching frame of frame t as:

$$M(t,0) = r$$

$$M(t,n) = \underset{f \in N(t,n)}{\arg\max}\, \mathrm{NC}(I(f), I(t))$$

where N is a set of tested frames where we set $N = [t + nb - 2, t + nb + 2]$ where b is the expected distance between frames corresponding to the same cardiac phase.[3] For $n > 0$ we call the matching frames forward, and for $n < 0$ backward.

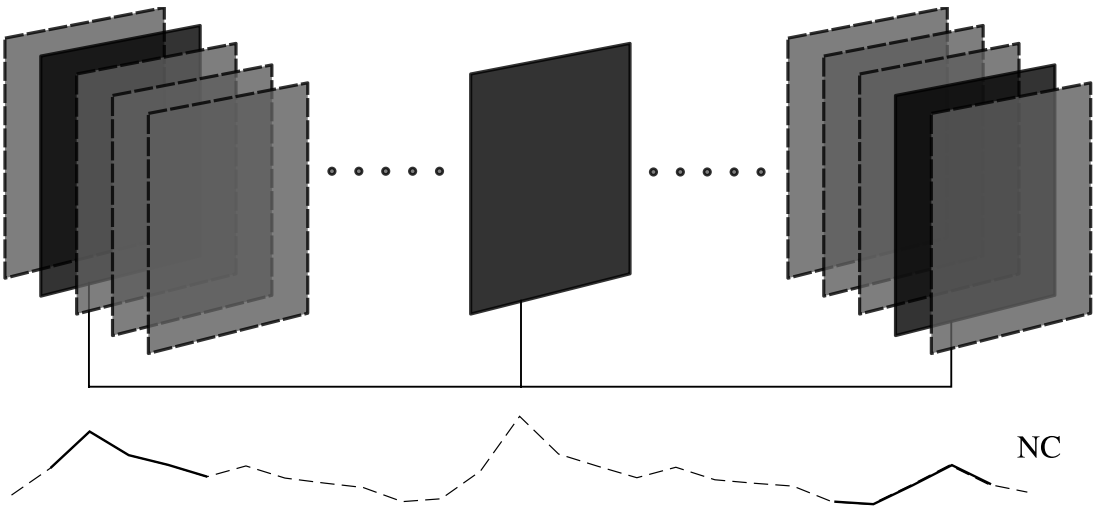


Figure 6.1: Matching frames are selected by a normalised correlation with the reference frame. A frame number of a matching frame is estimated from heart beat and the best matching frame is a frame close to estimate with maximum NC.

Heart rate detection

Selecting matching frames poses a question how to setup the constant b. If we had the synchronised ECG data, than clearly b should be derived from it, but as we lack ECG we need to detect the heart rate from the images, as precisely as possible. A physician can clearly see how a periodicity in a sequence is connected to the heart rate. We searched how this information could be extracted. As this approximation could not be compared to actual ECG, we can only speculate how good it is.

To detect the heart rate we follow the changes of overall intensity in the frame. The idea behind this is that movement of tissue in and out of the field of view produces a curve $r(t)$ with a frequency corresponding to the heart rate:

$$r(t) = \int I(\vec{x}, t)\ \mathrm{d}\vec{x}$$

Fourier transform (FFT) is applied to r and the highest peak $\hat{f}$ is found:

$$\hat{f} = \arg\max_f \mathcal{F}_t(r(t))(f) = \arg\max_f R(f)$$

However, the right frequency is recovered only if we neglect the oscillations inside of a cardiac cycle. One option is to use smoothing over approximately one heart beat

[3] We also tested an alternative definition, we call *linked matches* M_l which uses a different approach searching in $N = [M_l(t, n-1) - 2, M_l(t, n-1) + 2]$ for $n > 0$, and for $n < 0$ using $M_l(t, n+1)$ analogically.

(Figure 6.2), as follows:

$$s(t) = r(t) - (G * r)(t)$$

where G is a Gaussian, and apply FFT to s. Using linearity of the transform:

$$\hat{f} = \arg\max_f S(f) = \arg\max_f R(f)\,(1 - \mathcal{F}_t(G(t))(f))$$

Another reason why this approach works is because of non-rigid deformation that stretches and compresses the tissue periodically increasing and decreasing the overall intensity.

As a way to validate the heart rate estimate, we can assume that it should be approximately between 50 and 200 beats per minute (bpm) which means 0.83–3.33Hz, or 15–3.7 frames per contraction with sampling frequency of 12.5Hz. The high bound of 200bpm is probably overshoot as it is the maximum heart rate in patients [85]. With our data we get values corresponding to about 10 frames per contraction.

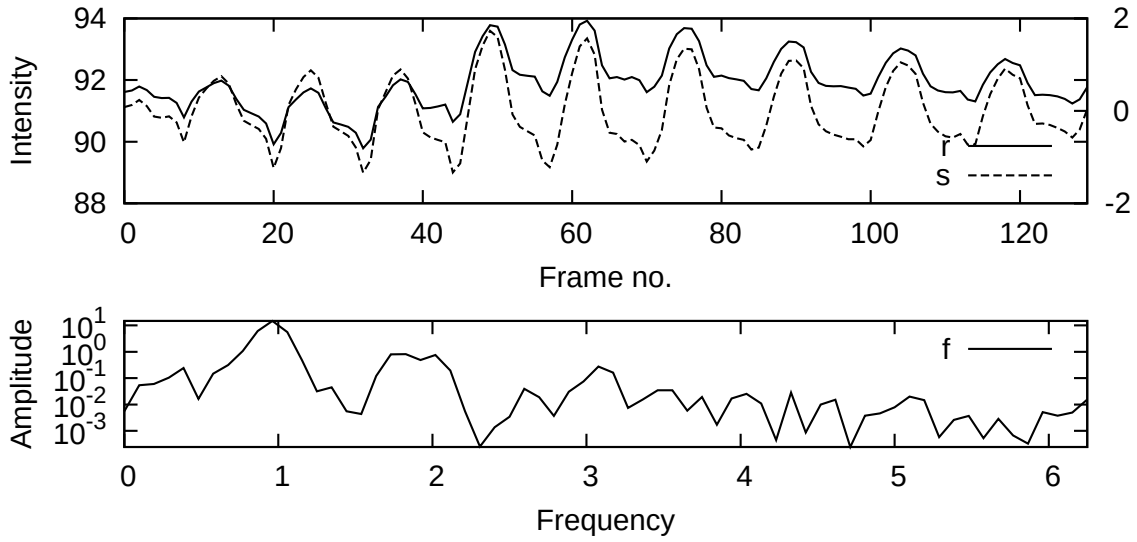


Figure 6.2: Original series r along with its smoothed counterpart s (top) and frequency spectrum of s with a logarithmic scale (bottom).

We also tried to apply this approach to sub-blocks of the frame. Sub-blocks has the advantage of providing more precise curves, as well as other information. In series of some blocks there is apparent diaphragm movement (Figure 6.3) This seems as a viable alternative to respiration detection method by Condurache [24], although we did not study this possibility in detail.

Background estimation

In computer vision, it is quite common to use background subtraction method to segment moving regions from background [31]. Usually a model for each object class

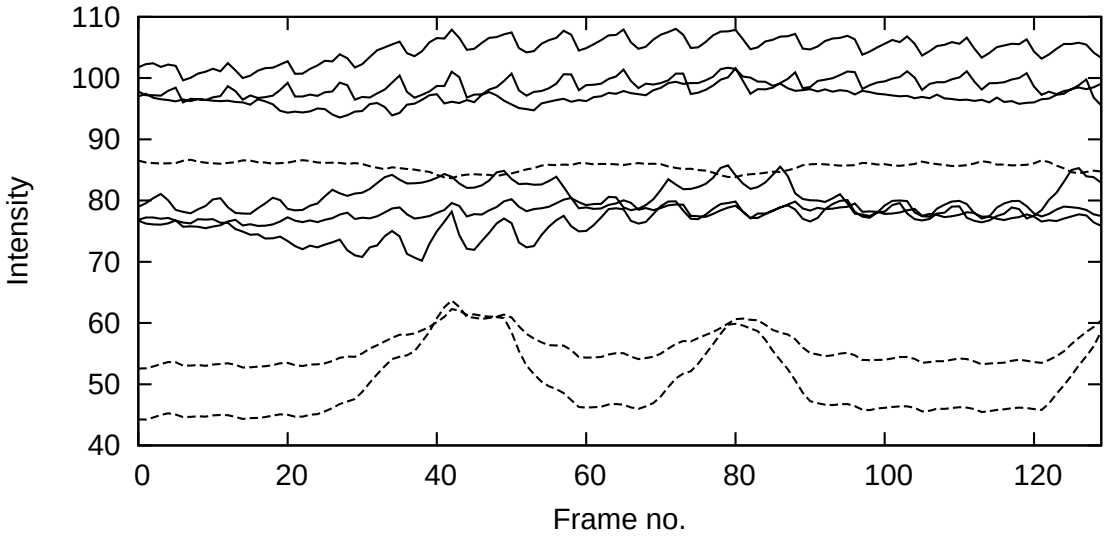


Figure 6.3: Heart rate curves r for nine sub-blocks. In 6 out of 9 series we can clearly identify the heart rate (solid). Remaining 3 series which correspond to the lower right corner of the image contain this frequency as well but also contain a strong signal due to diaphragm movement as a result of respiration (dashed).

in the scene is created and it is supposed that the intensity has normal distribution or eventually multi-modal normal distribution. Then the moving regions are detected as points where intensity changes abruptly. Some key assumptions in this approach do not hold for our data, such that we do not want to include neither parts of image where there is opacification nor a fixed segment of an artery into background. Because of these reasons it is improper to apply a method like this.

It is not straightforward to define the background. But as DSA works quite successfully, we try to define the background anyway. We use multiple frames and assume that a background value at one position $I(\vec{x}, t)$ is a random variable in t with a certain distribution. We can use the expected value $E[I(\vec{x}, t)]$ and suppose that the distribution is close to normal. Concerning the imaging process, this is reasonable for pixels without opacification, as both the quantum and structure mottle (statistical non-uniformities in X-ray detection) can be described as white noise [10, 37] and is successfully removed by averaging [92]. We define background B as a mean of all aligned frames:

$$B(\vec{x}, t) = \frac{1}{n} \sum_{f=0}^{n} I\left(T_{t,f}(\vec{x}), f\right)$$

where $I(\vec{x}, t)$ is a reference frame and $T_{t,f}$ is a transformation from $I(\vec{x}, t)$ to $I(\vec{x}, f)$.

In our experience, the subtracted version of the frame is always more readable as it highlights the opacified arteries and areas in general, see Figure 6.4. However, at the same time it introduces certain bias towards low opacity near strongly opacified moving objects such as arteries. This is caused by the fact such an object increases the values in their vicinity due to imperfect alignment. To remove this effect we can add

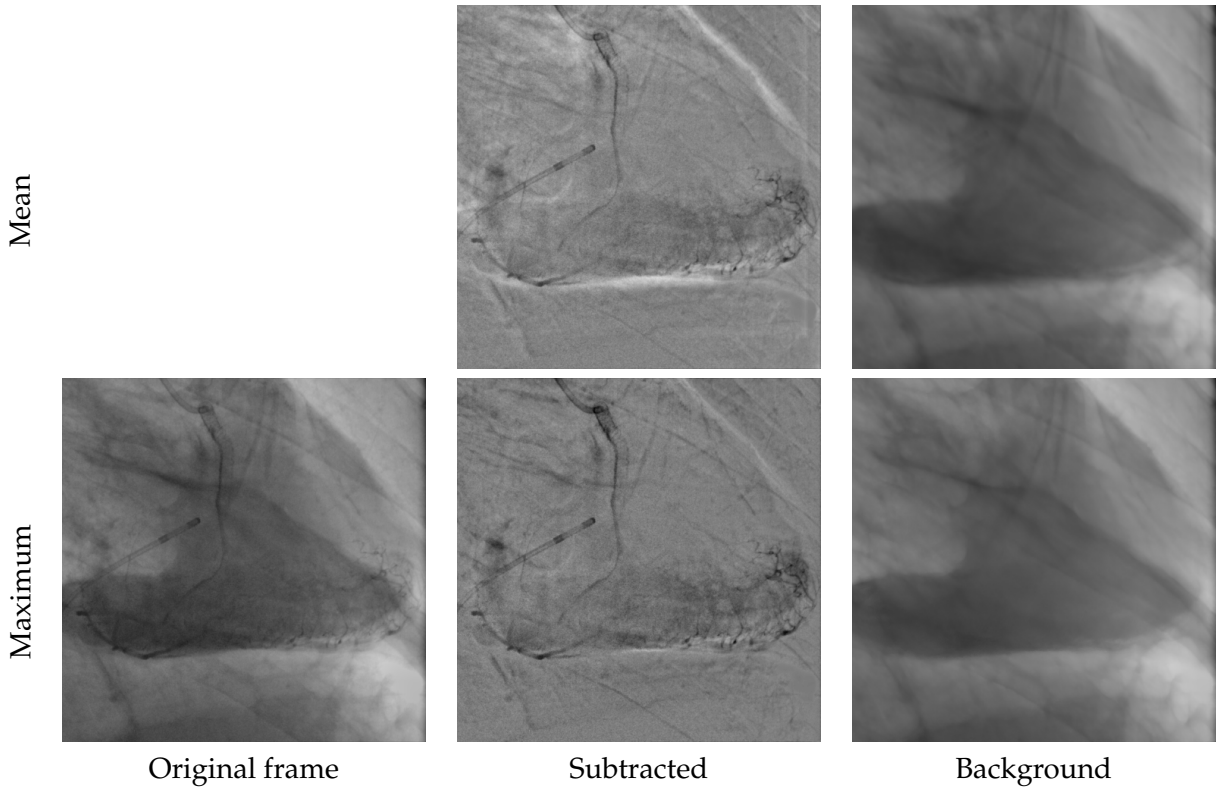


Figure 6.4: Background subtraction — compare how background using maximum is sharper as there is no averaging.

intensity weighing by vesselness:

$$B(\vec{x},t) = \sum_{f=0}^{n} \frac{1}{1+V(\vec{x},t)} I(T_{t,f}(\vec{x}),f) \Bigg/ \sum_{f=0}^{n} \frac{1}{1+V(\vec{x},t)}$$

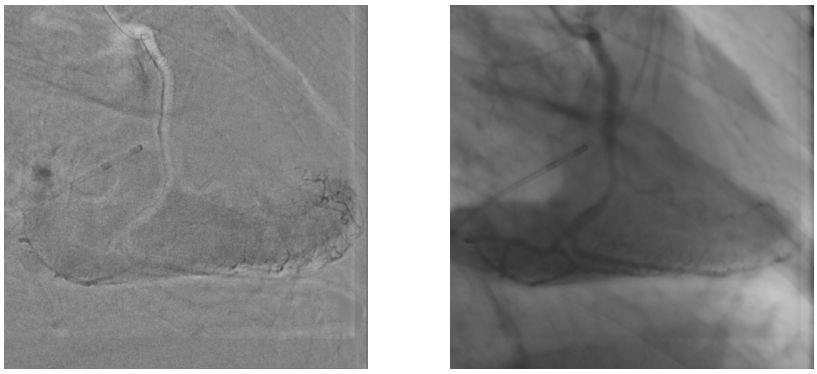
where $V(\vec{x},t)$ is a vesselness of the frame $I(\vec{x},t)$. Another option is to exchange mean for another measure, one that gives good results is maximum. Maximum estimated backgrounds usually contain more noise but in some aspects can be superior to mean estimated ones. This is especially due to the fact that no blurring happens and that no strong assumption about intensity distribution is made, see Figure 6.4.

One more thing we want to emphasise is that background image depends on the reference frame as all contributing frames are first transformed to coordinates of a reference frame.

A better approach to background estimation is to use only a subset of all frames, namely the matching frames from Section 6.2.1.

$$B(\vec{x},t) = \frac{1}{|M(t)|} \sum_{f \in M(t)} I(T_{f,t}(\vec{x}),f)$$

where $M(r)$ is a set of matching frames selected for background estimation. Mean estimated background is better focused/de-blurred compared to the one obtained from



(a) subtracted (b) background

Figure 6.5: Background subtraction with matching frames is more specific than with no special selection of frames, compare with Figure 6.4.

all frames, see Figure 6.5. The advantage is not only better quality but also significantly shorter computation time.

Further we can select only a subset of matching frames depending on the angiographic phase we want to obtain background for. This way we can extract arterial background to be able to identify vessels, see Section 6.2.2, or a background consisting of only the several first matching frames which contain minimum opacification but unlike for subtracting the first frame only (which is equivalent to DSA), it partially removes noise. Using all matching frames is questionable as in case of long sequence and long persisting opacity, we would include the opacity into background. The same happens if we use the first few frames and there is residual opacity from previous injection. A good set of matching frames for background subtraction depends on the information we expect to get and a single set alone might not provide the proper insight.

6.2.2 Opacity curves

Next, we define and analyse curves describing opacification in a given area, we refer to them as *opacity curves*. The most important curve of interest is ROI opacity curve, to be able to measure it, we have to select the ROI.

ROI selection

Let's consider how to define a ROI that is checked for opacity changes. We take approach similar to that of Condurache et al. [25] and start from a user-defined ROI $\Omega(t) \subset \mathbb{R}^2$ in a single frame t. This region is then automatically tracked throughout the sequence. Tracking from one frame to another is accomplished by warping $\Omega(f)$ from

a frame f to a frame t:[4]

$$\Omega(t) = T_{f,t}(\Omega(f))$$

The source frame f is selected so that matching frames (Section 6.2.1) are preferred over other frames to use the most precise transformations. Search for a source frame is two-pass, in the first pass we scan matching frames, in the second pass all frames. User interaction is indispensable to correct for the outcome of automatic process. In each pass user-defined ROIs are preferred over automatic ROIs thus lowering the need for user interaction.

ROI opacity curve

We construct an opacity curve q as a time course of ROI intensities after subtracting estimated backgrounds:

$$q(t) = \int_{\Omega(t)} B(\vec{x}, t) - I(\vec{x}, t) \, \mathrm{d}\vec{x}$$

The sign is changed to reflect that when we speak about high opacity when there are low intensities in input data. Similar to vesselness weighing in background estimation, we can deal with erroneously increased opacity by regarding only positive values, that is by integrating over region $\Omega'(t)$:

$$\Omega'(t) = \Omega(t) \cap \{\vec{x} \mid B(\vec{x}, t) - I(\vec{x}, t) > 0\}$$

In the place of mean using high percentile makes good sense as well because we are interested in the high part of histogram of $B(\vec{x}, t) - I(\vec{x}, t)$. 80-100 percentiles are reasonable choices.

Arterial opacity curve

Apart from ROI curve q we need an arterial opacity curve, as both are used by the perfusion model defined in the next section. Let us denote the two curves q_1 and q_2, the first being arterial opacity and the latter ROI opacity as they are used in the model in this order. Further, we describe how the q_1 curve is created.

We compute a weighed mean of values after background subtraction:

$$q_1(t) = \int (I(\vec{x}, t) - B(\vec{x}, t)) \, W(\vec{x}, t) \, \mathrm{d}\vec{x} \Big/ \int W(\vec{x}, t) \, \mathrm{d}\vec{x}$$

So there are two important decisions when extracting the curve, these are the type of background B and the mask or weighing W that selects arterial pixels.

The weights $W(\vec{x}, t)$ should define how much the given pixel pertains to an artery. This is measured by vesselness, so we can use a vesselness measure of choice from

[4]For the ease of notation we suppose an extension of transformation T to subsets of $\mathbb{R}^2$ in an agreement with transformation of their characteristic functions, $A = T(B)$ iff $\forall \vec{x} \; \chi_{A(\vec{x})} = \chi_{B(T(\vec{x}))}$ where $A, B \subset \mathbb{R}^2$.

Chapter 4. We threshold the vesselness, as weighing by vesselness alone does not work well because it includes too many pixels where vesselness is small but non-zero. The best way of thresholding we found is to mask vesselness by thresholded top-hat of arterial background A:

$$\begin{aligned} W &= \chi(T > \tau)V \\ T &= A \bullet (C - A) \end{aligned}$$

where C is a circle structuring element (as wide as the thickest vessel). Here, top-hat filter works as a naive vesselness filter which is quick and works reasonably well. This way we fix the part of image that gets selected for all frames, and mask out the parts of image which are not at place where vessels are expected. Threshold τ was fixed to 0.1 with our data, generally it should be set to reject false positives.

Regarding the background subtraction, we use a non-arterial background, eg. a first match as it contains minimum opacity ensuring the resulting curve lies above zero.

One more thing to consider is that the scene is not constant. There are vessel-like objects that enter/leave the field of view during the sequence due to a shift of the table. Specifically, this happens for a catheter, and it leads to an artificial change of opacity. To limit these changes, we decided to take into account only the part of the scene that is contained in the last matching frame. We propagate this restriction of the scene to all other used frames using the recovered translations, meaning we crop the images to the region that maps inside their last matching frames. This way we maintain the same objects in the followed scene.

So, the final arterial curve extraction is defined as:

$$q_1(t) = \int_{S(t)} \left(I(\vec{x}, t) - B(\vec{x}, t)\right) W(\vec{x}, t) \, \mathrm{d}S(t) \Big/ \int_{S(t)} W(\vec{x}, t) \, \mathrm{d}S(t)$$

where W is weight image for a given frame t and S is a relevant part of the scene.

Perfusion model

The extracted opacity curves can be used to assess perfusion visually. In addition, we propose to use a mathematical model of perfusion and transition of contrast agent between arteries and ROI where parameters of the model will be connected to TMP grades. We use the classical toxicological model — a *compartment model* [14]. Such model consists of several compartments containing a substance which is transferred between the compartments. Usually a model includes a source and a sink connected to one or more compartments. Equations describing a compartment model are usually derived using a conservation law — the model is closed with the exception of the known sources and sinks. In other words concentration is changed only by transition from compartment to compartment, or by sources/sinks.

A concentration of the given substance is considered to be homogeneous in each of the compartments, so that the concentration in one compartment at given time can be expressed as a single number.

We will consider the compartment model shown in Figure 6.6 which is described

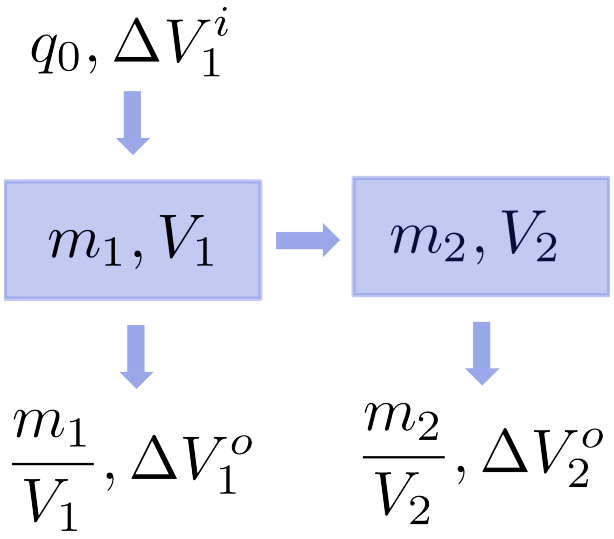


Figure 6.6: Two compartment model. Compartment 1 models arterial tree and compartment two is the ROI. A m_i is mass of the dye in compartment i, V_i volume of compartment i, ΔV_1^i is inflow rate of compartment 1 and ΔV_i^o outflow rate of compartment i, t_d is a delay between compartments, and finally q_0 is input concentration.

by the following system of differential equations:

$$\begin{aligned}\frac{\partial m_1}{\partial t}(t) &= q_0 \Delta V_1^i - \frac{m_1}{V_1}\Delta V_2^o - \frac{m_1}{V_1}\Delta V_1^o \\ \frac{\partial m_2}{\partial t} &= \frac{m_1(t-t_d)}{V_1}\Delta V_2^o - \frac{m_2(t)}{V_2}\Delta V_2^o\end{aligned}$$

We can derive an explicit solution for $m_i(t)$ if we define the input concentration q_0. Let q_0 be a step function:

$$q_0(t) = \begin{cases} 1 & \text{for } t \leq t_m \\ 0 & \text{otherwise} \end{cases}$$

Solving the system yields:

$$m_1(t) = \begin{cases} V_1 - V_1 e^{-at} & \text{for } t \leq t_m \\ \alpha V_1 e^{-a(t-t_m)} & \text{otherwise} \end{cases}$$

where

$$a = \frac{\Delta V_1^i}{V_1}$$

and

$$m_2(t) = \begin{cases} 0 & \text{for } t \leq t_d \\ V_2 + \frac{\Delta V_2^o}{a-d} e^{-a(t-t_d)} - \left(V_2 + \frac{\Delta V_2^o}{a-d}\right) e^{-d(t-t_d)} & \text{for } t \leq t_d + t_m \\ \alpha \frac{\Delta V_2^o}{d-a} e^{-a(t-t_d-t_m)} + \left(m_2^1(t_d+t_m) - \alpha \frac{\Delta V_2^o}{d-a}\right) e^{-d(t-t_d-t_m)} & \text{otherwise} \end{cases}$$

where

$$d = \frac{\Delta V_2^o}{V_2}.$$

$$t_m = -\frac{1}{a}\log(1-\alpha)$$

The value $m_2^1(t_d + t_m)$ in the last part of definition of m_2 is final value of the previous part of solution set so to maintain continuity. We can rewrite the solutions for concentrations $q_i = \frac{m_i}{V_i}$ as:

$$q_1(t) = \begin{cases} 1 - e^{-at} & \text{for } t \leq t_m \\ \alpha e^{-a(t-t_m)} & \text{otherwise} \end{cases}$$

and

$$q_2(t) = \begin{cases} 0 & \text{for } t \leq t_d \\ 1 + \frac{d}{a-d}e^{-a(t-t_d)} - \left(1 + \frac{d}{a-d}\right)e^{-d(t-t_d)} & \text{for } t \leq t_d + t_m \\ \alpha\frac{d}{d-a}e^{-a(t-t_d-t_m)} + \left(q_2^1(t_d+t_m) - \alpha\frac{d}{d-a}\right)e^{-a(t-t_m)} & \text{otherwise} \end{cases}$$

Again $q_2^1(t_d + t_m)$ serves maintaining continuity. The parameters of the model are a, d, α, t_d. Figure 6.7 shows concentrations for $a = 0.11, d = 0.1, \alpha = 0.9, t_d = 10$.

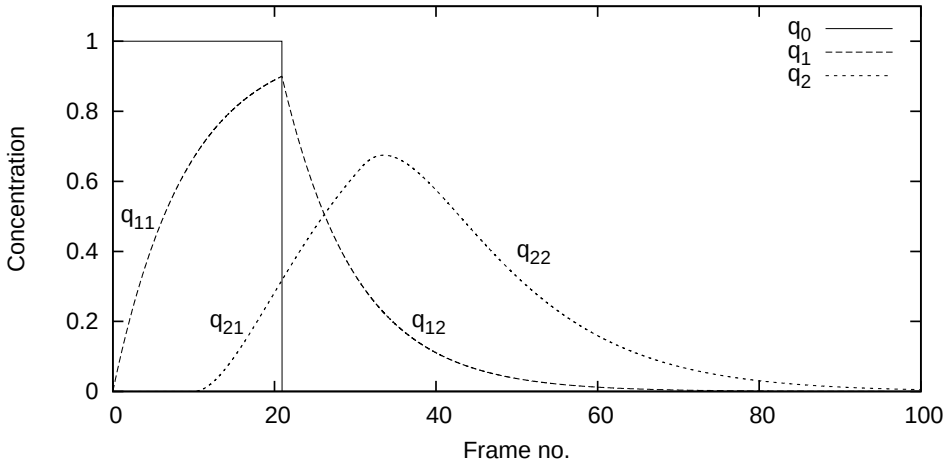


Figure 6.7: Concentrations q_1, q_2 in the two compartments and the injection concentration q_0. Both parts of solutions (q_{11}, q_{12} and q_{21}, q_{22}) are shown.

However, we have no means of recovering the true information about input concentration q_0.[5] So we use simpler one-compartment model where only the link between arterial curve q_1 and ROI curve q_2 is modelled. The parameters of a new model are delay t_d, and transmission d:

$$\dot{q}_2(t) = d\,(q_1(t - t_d) - q_2(t))$$

[5]The injection pattern is generally known [67] but the precise time course is not recorded.

d	t_d	*evaluation*
>0.1	L	TMP 3
0.01–0.1	H	TMP 2
0.001–0.01	H	TMP 1
<0.001	n/a	TMP 0

Table 6.1: The relationship of compartment model parameters and TMP levels. TMP levels can be estimated by transition parameter d. Higher values of t_d should be typical for TMP 1-2 but our experiments did not confirm it.

and we use the following discrete version:

$$q_2(t+1) = q_2(t) + [cq_1(t) - dq_2(t)]$$

The hypothesis is that low value of parameter d points to slow transition of contrast agent from arteries to capillaries and therefore to higher TMP level, TMP level with zero perfusion should have $d \approx 0$. Table 6.1 shows estimated values that decide about TMP levels, Figure 6.8 shows the effect of parameter d to our model. In the Section 6.3.5 we provide an experiment to support our claims.

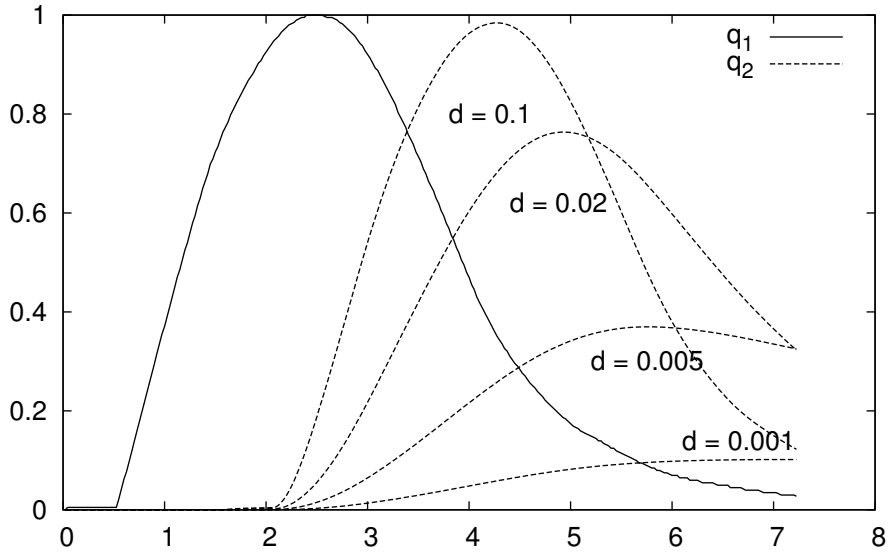
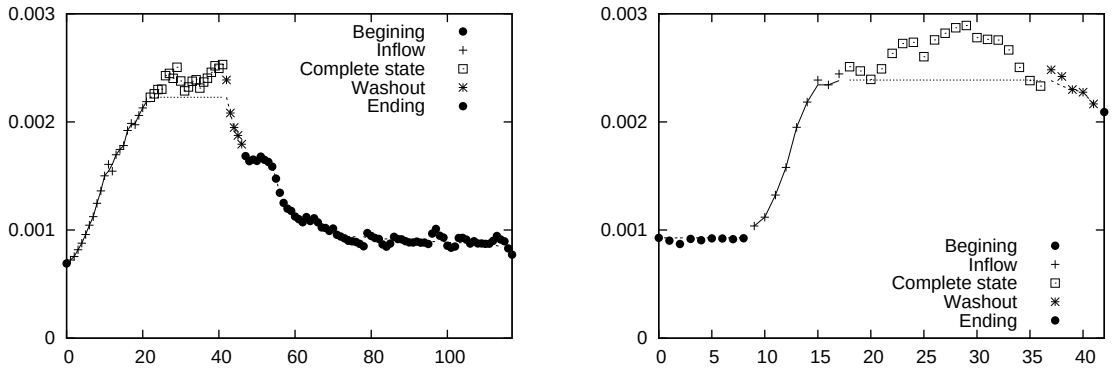


Figure 6.8: Two-parameter compartment model with various values of transition parameter.

6.2.3 Phases

Before finding an article on complete phase frames detection [2], we attempted to do phase detection, mainly with the aim of selecting the plausible sequences for further processing. All sequences should contain at least three heart-beat long part of the washout phase to make possible the assessment of TMP. However, the tests clearly reveal this is not the case with a majority of sequences.



(a) all phases (b) only beginning

Figure 6.9: Phases detection in two sequences, correct sequence with all phases present (a), and an example of unusable sequence with phases missing (b).

We can detect the phases if we follow the vesselness measure throughout the sequence. As high vesselness is by definition signalling opacification of arteries we can detect arterial phase as a part of sequence where overall vesselness rises, or stays still. Start of the washout phase is marked by a decrease of vesselness. Overall vesselness is computed as a sum of vesselness over a frame. This measure proved reasonably robust but must be further smoothed by median filtering with long kernel, at least two cardiac cycles. Smoothing can be justified by the fact we want to detect only the high rise of vesselness corresponding to arterial phase and nothing more. This way we filter out periodical changes caused by heart beat.

6.3 Experiments

As described in previous sections, most of the methods have many parameters (such as what kind of background subtraction to use) which where not precisely specified, so we discuss it here along with outcome of several tests.

6.3.1 Registration in background subtraction

First, we want to show the effect of registration on background subtraction. The need for registration is apparent from Figure 6.10 where we obtain a reasonably clear image only after motion compensation. This is caused partly by the fact that this sequence was recorded with patient breathing. On the other hand, we need not to warp frames if there are sufficient frames forming the background that averages out the differences, optionally along with vesselness weighing.

6.3.2 Metric in frame selection

Regarding frame selection, NC proved to be adequate metric but we can use MSE as well, as the results are identical, MI seems less usable, see Figure 6.11. As we have no

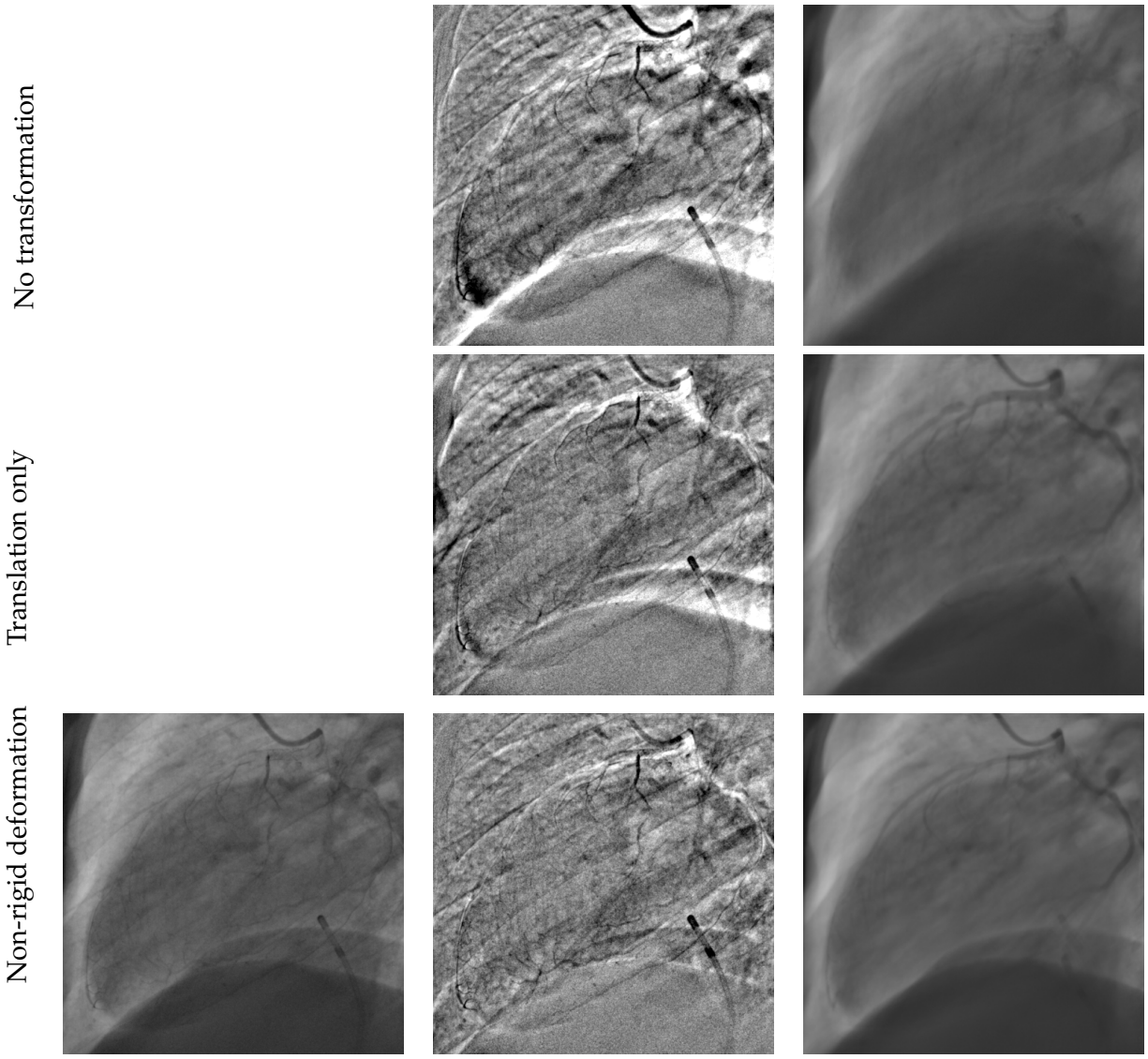


Figure 6.10: Effect of applying recovered transformation for background subtraction, note the misalignment of diaphragm, and errors in top left corner and near catheter that is corrected only with deformation.

ECG, we can only say that the peaks of all metrics match with our visual observation of the heart beat.

6.3.3 ROI tracking test

Next we tested ROI tracking, the ground-truth ROIs were input manually and each one of them was selected as the source mask which initialised the tracking. We measured

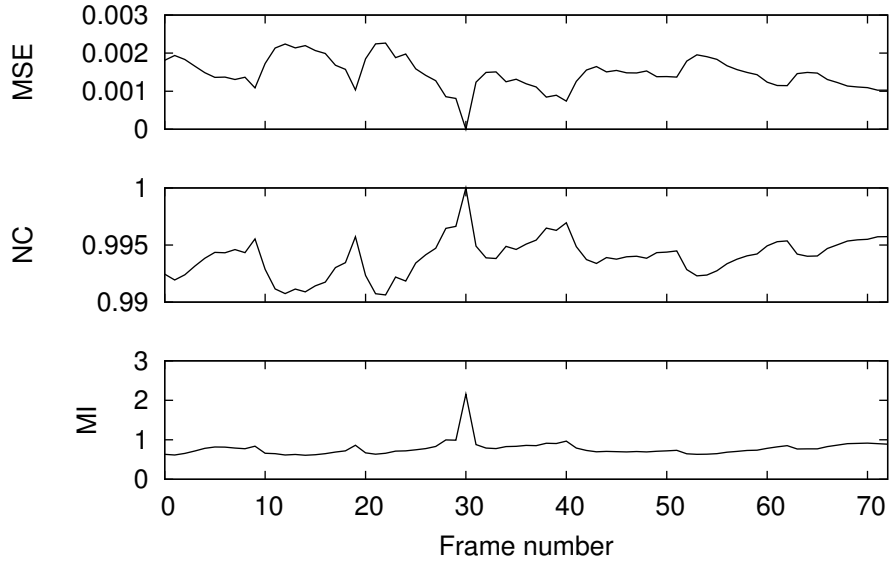


Figure 6.11: Effect of choice of metric to search of matching frames is negligible. Only the MI maximum is less pronounced.

the overlap between ground-truth ROI Ω_0 and estimated ROI Ω as[6]:

$$\frac{|\Omega_0 \cap \Omega|}{|\Omega_0 \cup \Omega|}$$

Our tracking reaches overlap of $59\% \pm 6$ to $72\% \pm 5$ depending on the size of the ROI[7] Especially hard is tracking ROI between frames with different cardiac phase, so we tested the improvement of tracking when initialised along one heart beat, results are significantly better reaching overlap of 84 ± 7 and 89 ± 3, see Figure 6.12 as well. The overlap below 50 percent is only exceptional which is a bound that was used by Condurache as a measure of success. Overlaps for the three curves shown in the Figure 6.12 were 0.81, 0.88 and 0.91.

We also implemented an alternative ROI tracking by Condurache [25] but this method did not prove to be usable with our data. Probable reason is insufficient robustness. Detected feature points are not matched correctly due to an ambiguity of their description by similarity which is high for near feature points of similar shape, see Figure 6.13. Such pairs produce too large or too small arbitrary direction displacement. Also the tracking takes more time and as it is dependant on input ROI, the results cannot be precomputed as in our approach.

6.3.4 Arterial curve extraction parameters

To select between the options for the two important components of arterial curve extraction (mask and background, see Section 6.2.2), we visually compare the obtained

[6]Lately, we have found a similar measure $\frac{2|\Omega_0 \cap \Omega|}{|\Omega_0|+|\Omega|}$ is used to compare segmentation results [115].

[7]It seems that the error is not linearly dependent on ROI size as the overlap measure supposes. Regarding the overlap, large ROI is tracked better than small.

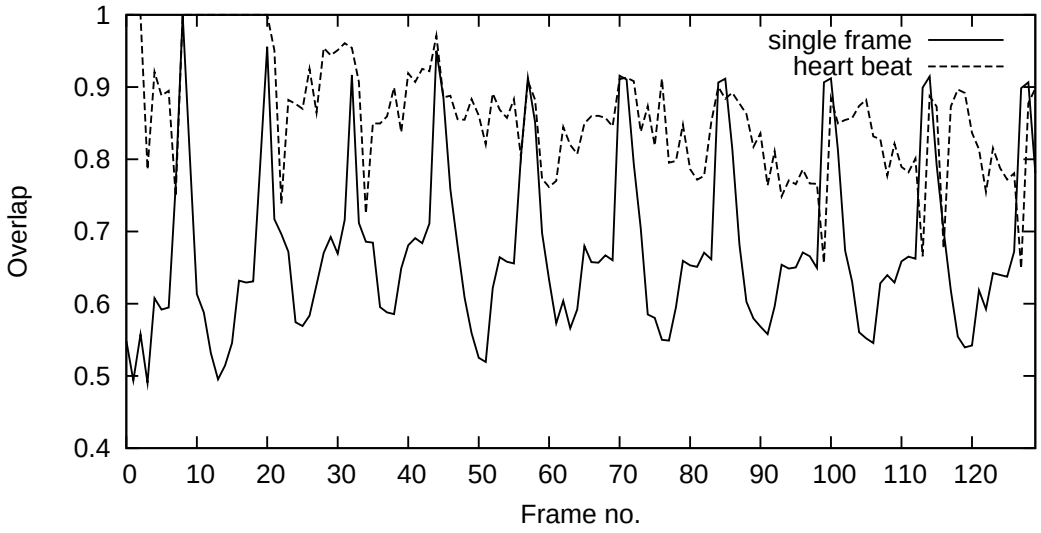


Figure 6.12: Percentage of ROI overlap for tracking results with initialisation in one frame and in all frames along one heart beat. We can see that single frame initialisation yields high overlap particularly for its matching frames. Much better tracking is obtained while initialising along one heart beat.

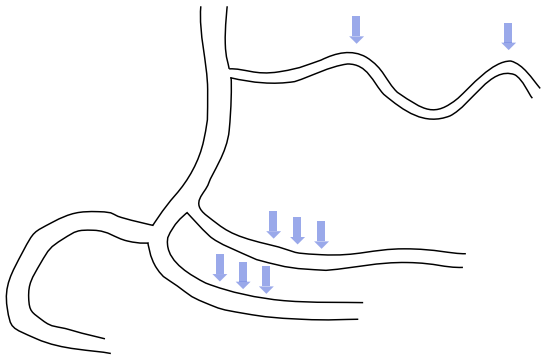
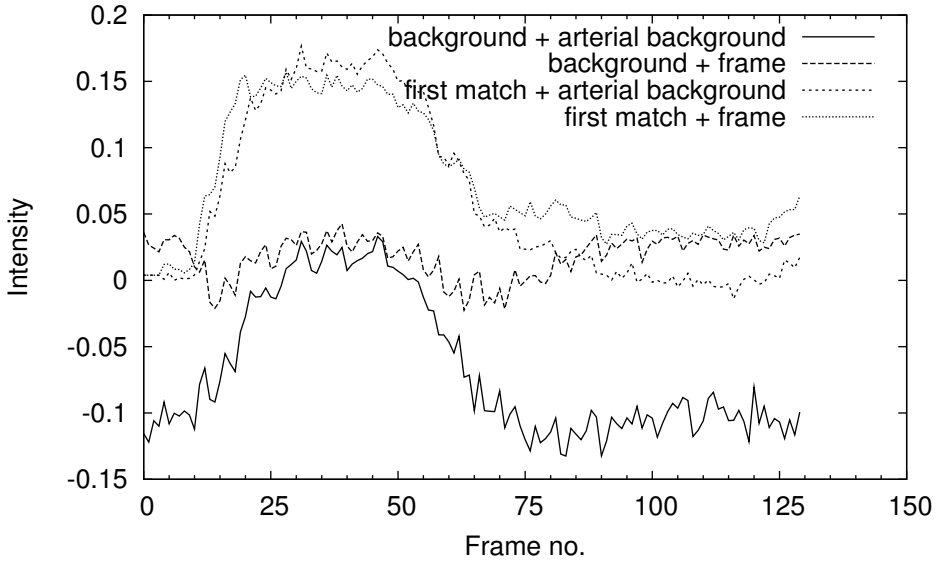


Figure 6.13: ROI tracking based on feature points fails with vessels which have several similar feature points near to each other.

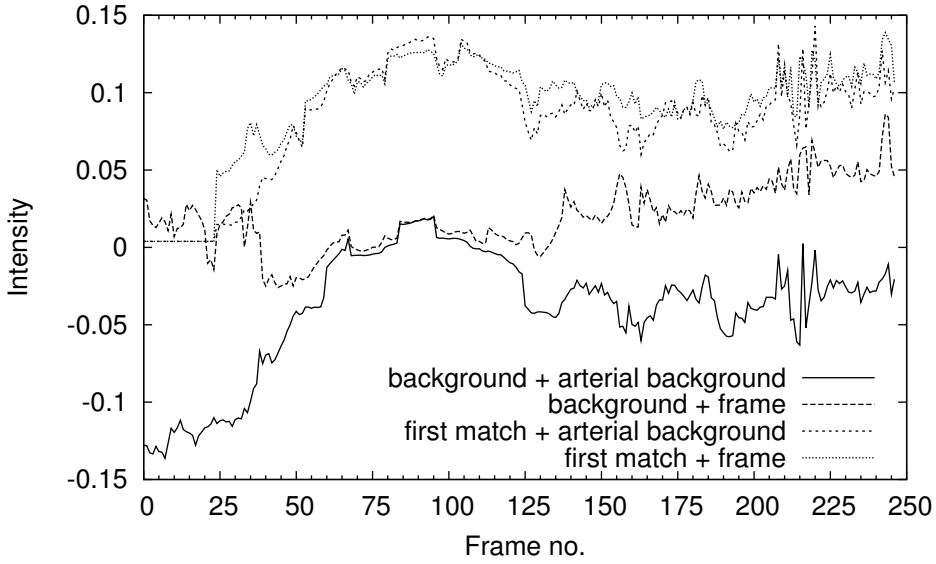
curves. We can see that using the actual frame as a mask can incorrectly increase values in the end of the sequence when there is no opacity, while opacity curves obtained with arterial background, correctly fall down, see Figure 6.14(a).

In Figure 6.14(b), we can see one of the problems even with arterial background and first match version of extraction, as the curve should fall down when opacified blood leaves arteries. This does not happen as in this sequence some opacification persists under/above the arteries and due to a projective character of X-ray increase the values in the last part of the curve. We think this effect could be removed if we detect arterial phase by other means and mask out the parts outside arterial phase.

As we use first match subtraction in arterial curve extraction, we decided to use it



(a) successfully extracted opacity curve



(b) opacity curve with erroneous final part

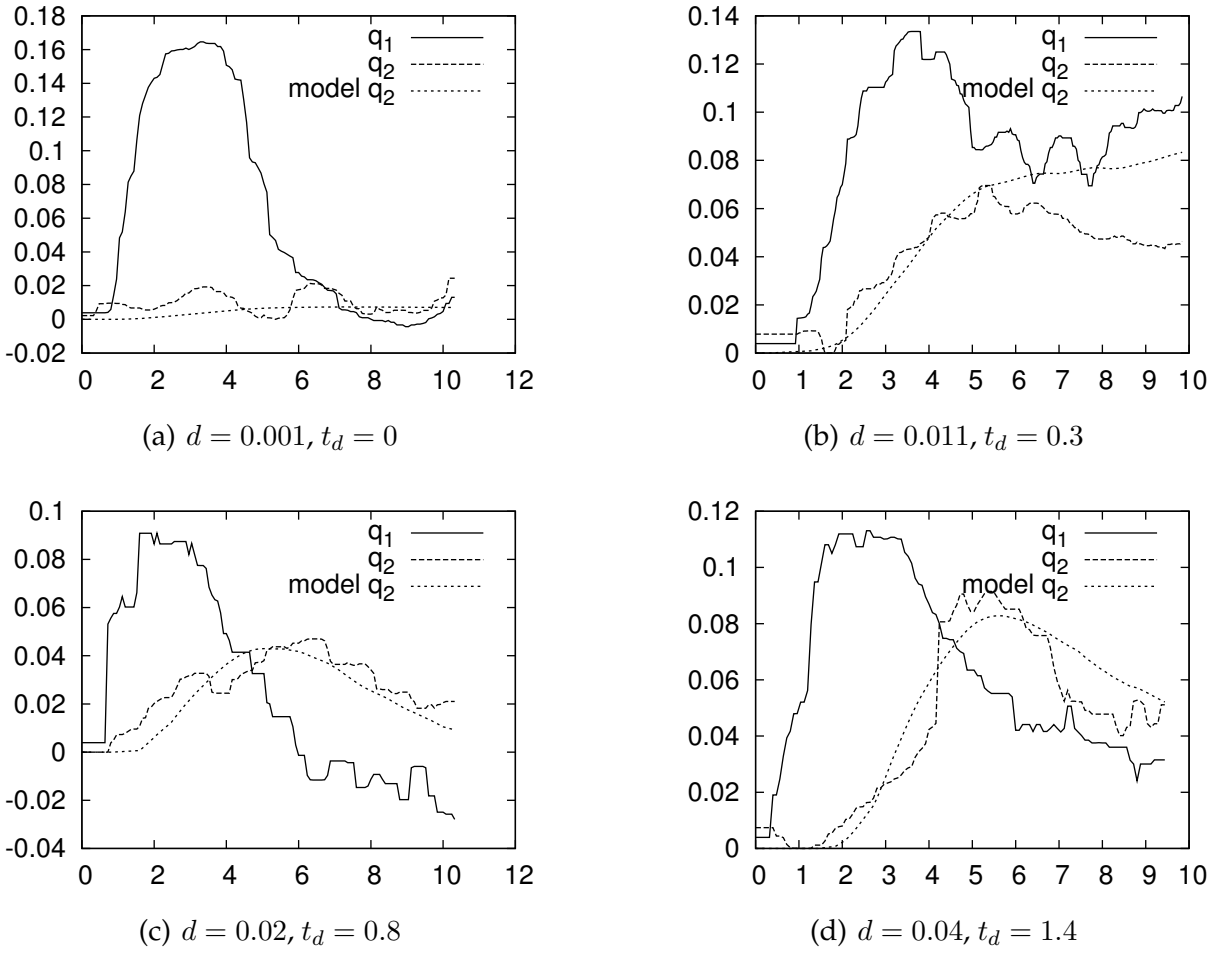
Figure 6.14: Arterial opacity curves depend on the correct selection of a subtracted background + arterial mask.

in opacity curve extraction as well because only this way we get comparable values for both curves.

6.3.5 Compartment model test

We applied our compartment model to real angiograms. The model was fitted to the data in the least-squares sense and using an exhaustive search through the parameter domain. Exhaustive search is a good choice in this case as it can provide sufficient

precision in less than a second. The arterial and ROI opacity curves were preprocessed by a median filter to remove outliers. Figure 6.15 shows selected values obtained with four tested sequences with different levels of opacity. Transition parameter d provides reasonable values, on the other hand delay parameter t_d does not. One of the reasons why the delay parameter does not give any useful results is that we consider arteries to be a single compartment which implies an assumption of homogeneity while in reality arteries are not completely homogeneous.



(a) $d = 0.001$, $t_d = 0$

(b) $d = 0.011$, $t_d = 0.3$

(c) $d = 0.02$, $t_d = 0.8$

(d) $d = 0.04$, $t_d = 1.4$

Figure 6.15: Real input data from four patients, in (a) the model does not detect any opacification, in (b) opacification is slow, in (c-d) opacity persists a long time.

6.4 Conclusion

The method developed to enhance angiograms improves readability of angiograms. Non-rigid deformation is not always needed especially with background created out of sufficient number of frames where possible differences average out, but they improve the results with single frame background and especially sequences where the heart beat is a non-integer multiple of sampling frequency.

ROI tracking works well enough to produce usable ROI opacity curve. The actual tracking is quite fast when the deformations are precomputed, as only the source frame must be selected and the ROI warped.

To prove the relevance of the method classification results obtained from the compartment model should be tested with a large number of sequences. We did not have enough input sequences to do this but the results are in good agreement with visual inspection. The same goes for extracted opacity curves.

Chapter 7

Implementation

In this chapter, we briefly describe programming tools and libraries used in implementations of the code associated with this thesis. Part of the implementations (testing) use Matlab and its Image Processing toolbox, the other part (applications) was implemented using C++ and third-party libraries.

7.1 Application

We implemented two main applications and several auxiliary tools. Here we would like to write only few remarks about `qtx`, a graphical user interface to opacity quantification and `clx`, a command line application that precomputes several things to improve the responsiveness of `qtx`. All registrations, heart rate detection and frame matching are performed in `clx`. `qtx` serves as a means to view sequences and difference sequences with enhanced opacity. `qtx` is also used to show plots of opacity in ROI and in local neighbourhood. ROI is selected in `qtx` as well. More precise instructions on installing and using developed applications is provided in user's manual found on the provided CD.

7.2 Overview of used libraries

In implementing our applications we used the following third-party libraries. Proper setup of these libraries is described in user's manual, here we only briefly introduce each of them:

Insight toolkit (ITK) `[http://www.itk.org/]` is aimed at registration and segmentation tasks, it simplifies implementation of complicated registration algorithms. We use the ITK for all our implementations of registration code and also image manipulation, and for our implementation of multiscale vesselness filter. Also the ITK software guide [49] serves as a good reference not only to ITK but also to image registration and segmentation in general. We contributed back patches of the ITK code we used and plan to contribute some of the filters developed during the work on this thesis.

vxl library [http://vxl.sf.net/] is used for optimisation routines that encapsulate in fact the the versions of the classical numerics library Netlib, above all we use the limited-memory BFGS optimiser. Equation solvers in non-rigid ICP are taken from the vxl library, as well.

GDCM library [http://www.creatis.insa-lyon.fr/Public/Gdcm/] is used to manipulate DICOM input, reading and interpreting the data.

libkdtree [http://libkdtree.alioth.debian.org/] is an alternative kd-tree implementation to ITK's one and was used to test non-rigid ICP. Although its performance is comparable to ITK, the usage is more straightforward because it follows standard SGI's STL container style.

alglib [http://www.alglib.net/optimization/] is used as a source for alternative C++ implementation of L-BFGS and LM optimisation to check implementations provided by ITK.

CImg library [http://cimg.sf.net/] is used as source for Tschumperle-Deriche's inpainting algorithm.

Qt toolkit [http://trolltech.com/products/qt] is used for graphical user interface (GUI) implementation.

7.3 Notes to DICOM reading

We encountered two issues in implementation of DICOM reading: first, representation of intensity values, and second, correct timing.

Intensity values with XA/XRF modality come with look-up table of intensity values but it is specified that this table maps values saved in DICOM file to beam intensities, DICOM tag *(0028,3000) ModalityLUTSequence*. DICOM tags *(0028,0103) PixelRepresentation* and *(0028,1040) PixelIntensityRelationship* refer to this reverse mapping. This can cause confusion but no mapping has to be done in order to display the frames, values saved in file are ready to be displayed.

As we deal with multi-frame files, we have to decide about frame rate at which to display individual frames. In some DICOM files there is a tag which defines recommended frame rate but our data does not contain it. Apart from this, there are *(0018,1063) FrameTime*, *(0018,1066) FrameDelay* and *(0028,0009) FrameIncrementPointer*. Real time in milliseconds of a frame n is given as:

$$t_n = FrameTime + (n-1)FrameDelay$$

in the case that FrameIncrementPointer points to FrameDelay which is the case with our input files. Therefore, the frame rate is fully specified by *(0018,1066) FrameDelay*.

Chapter 8

Summary

8.1 Conclusion

In this thesis we presented the results of our work on processing angiographic sequence data with the aim of opacity quantification. We reviewed and implemented several methods for vessel extraction that either provide vesselness measure or centerline. The results of this preprocessing step are used in next stages of our algorithm. We reviewed and implemented several methods for image registration, showed that with given data feature-based registration is not feasible and exchanged it for robust image-based approach that aligns images using hierarchical B-splines. Careful parameter tuning of used limited-memory BFGS optimiser was done.

Finally, we thoroughly searched angiography-specific literature for opacity quantification and to our surprise did not find any method that solves it. We suggested one such method with many possible variants that mimics classical DSA for our cardiac data. We implemented an application that enables tracking a ROI in a sequence using registration results and displaying opacity either as enhanced sequences or as time series in a plot.

The main contribution of this thesis lies in review and testing of the methods suitable for the task of opacity quantification, as well as, proposing a method to enhance and extract opacity from angiographic data.

8.1.1 Future research

Given the allotted time, we failed to deliver an algorithm that would give TMP or MBG levels automatically. We would need more labelled data for this purpose. However, we propose several building blocks that can lead towards the ideal algorithm. Registration of near and far frames (in terms of cardiac cycle phase) presents different problems, for the purposes of opacity evaluation image registration of close frames is needed, and the registration must be intensively regularised not to smooth out opacity changes. On the other hand, for the purposes of ROI estimation, regularisation should be less intensive to be able to align even far frames. Designing specialised methods for each of these tasks can be expected to improve the results, above all ROI estimation. Registration itself is difficult because of the projectional distortion of the data, the ribcage in the

background and the electrode in foreground that move differently from the soft tissue of the heart. We think electrode segmentation and bone masking could further improve the results.

One of the main inadequacy of our data is that the ECG data is missing. Many angiographic machines are now coupled with ECG. If the ECG data was available, it would seem worthwhile to reconsider some of the approaches, such as feature-based registration,

At last, to properly validate the compartment model outcome, we would need to test with a statistically significant number of sequences.

8.1.2 Image acquisition

We would like to comment on the image acquisition process. Data used throughout this thesis was obtained prior to the idea of using any automatic computer method to process it. That is why the data does not fullfil all the requirements we would like it to. To seriously exploit the techniques proposed in this thesis one should consider some changes in image acquisition.

We found following items to be the most important factors:

1. electrode should be removed from the field of view
2. fixed view, ie. no sliding
3. all phases should be present

The rationale for all of these items is clear, the first two are to increase the chance for a good alignment and the third one to actually assure there is the vital data we need. However, we understand that the first one and the last condition cannot be always adhered to, the first because of legal matters and the last one to limit the radiation dose.

Higher time resolution would certainly also improve the performance. The 12.5 frames per second we have used, seems to be the lowest possible value as changes between frames are in some cases significant — movement of arteries is really abrupt (up to 30 pixels between consecutive frames), making the deformable registration difficult. Note that with this frame rate the heart beat can contain as few as six frames, for patients with high heart rates. More common frame rates are 25 or 30 fps nowadays which sounds more appropriate.

And the last problem that complicated the analysis was a small range of intensities (8bit with noise) where moreover only a small part of the 256 levels available were used.

Bibliography

[1] Yezzi A., *Modified curvature motion for image smoothing and enhancement*, IEEE transaction on image processing **17** (1998), no. 3, 345–352.

[2] T. Aach, A. Condurache, K. Eck, and J. Bredno, *Statistical-model based identification of complete vessel-tree frames in coronary angiograms*, Electronic Imaging 2004: Computational Imaging II (San Jose, USA) (C. A. Bouman and E. L. Miller, eds.), SPIE Vol. 5299, January 18–22 2004, pp. 283–294.

[3] T. Aach, I. Stuke, C. Mota, and E. Barth, *Estimation of multiple local orientations in image signals*, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) 2004 (Montreal, Canada), vol. III, May 17-21 2004, pp. 553–556.

[4] D. Anguelov, D. Koller, P. Srinivasan, S. Thrun, H-C. Pang, and J. Davis, *The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces*, Advances in Neural Information Processing Systems (NIPS 2004) (Vancouver Canada), 2005.

[5] S. Osher ans J.A.Sethian, *Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations*, Journal of Computation Physics **79** (1988), 12–49.

[6] R. Anxionnat, S. Bracard, X. Ducrocq, Y. Trousset, L. Launay, E. Kerrien, M. Braun, R. Vaillant, F. Scomazzoni, A. Lebedinsky, and L. Picard, *Intracranial aneurysms: Clinical value of 3D digital subtraction angiography in the therapeutic decision and endovascular treatment*, Radiology **218** (2001), 799–808.

[7] National Electrical Manufacturers Association, *Digital imaging and communications in medicine (dicom)*.

[8] G. Aubert and P. Kornprobst, *Mathematical problems in image processing: Partial differential equations and the calculus of variations (second edition)*, Applied Mathematical Sciences, vol. 147, Springer-Verlag, 2006.

[9] S. K. Balci, P. Golland, and W. M. Wells, *Non-rigid groupwise registration using B-spline deformation model*, MICCAI Open Workshop (2007), 173–190.

[10] G.T. Barnes, *Radiographic mottle: A comprehensive theory*, Medical Physics **9** (1982), 656–667.

[11] M. Bertalmío, G. Sapiro, and G. Randall, *Morphing active contours*, IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000), no. 7, 733–737.

[12] P.J. Besl and N.D. McKay, *A Method for Registration of 3-D Shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence **14** (1992), no. 2, 239–256.

[13] A. Bhuiyan, B. Nath, and J. Chua, *An adaptive region growing segmentation for blood vessel detection from retinal images*, VISAPP (1), 2007, pp. 404–409.

[14] M. Blomhøj, T.H. Kjeldsen, and J. Ottesen, *Compartment models*, Available at `http://www4.ncsu.edu/~msolufse/Compartmentmodels.pdf`, 2005.

[15] F.L. Bookstein, *Principal warps: Thin-plate splines and the decomposition of deformations*, IEEE Transactions on Pattern Analysis and Machine Intelligence **11** (1989), no. 6, 567–585.

[16] J. Brieva, E. Gonzalez, F. Gonzalez, A. Bousse, and J.J. Bellanger, *A level set method for vessel segmentation in coronary angiography*, Engineering in Medicine and Biology Society, Proceedings of 27th Annual International Conference of the IEEE, 2005, pp. 6348–51.

[17] E. Bullitt and S.R. Aylward, *Analysis of time-varying images using 3d vascular models*, Proceedings of Applied Imagery Patt. Recog. Works., 2001, pp. 9–14.

[18] P. Čech, A. Andronache, L. Wang, G. Székely, and P. Cattin, *Piecewise rigid multimodal spine registration*, pp. 211–215, Springer, 2006.

[19] T.F. Chan and L.A Vese, *Active contours without edges*, IEEE Transactions on Image Processing **10** (2001), no. 2, 266–277.

[20] H. Chui, *Non-rigid point matching: Algorithms, extensions and applications*, Ph.D. thesis, Yale University, May 2001.

[21] H. Chui and A. Rangarajan, *A new point matching algorithm for non-rigid registration*, Computer Vision and Image Understanding **89** (2003), 114–141.

[22] A. Condurache, T. Aach, K. Eck, and J. Bredno, *Fast detection and processing of arbitrary contrast agent injections in coronary angiography and fluoroscopy*, Bildverarbeitung für die Medizin, 2004, pp. 5–9.

[23] A. Condurache, T. Aach, K. Eck, J. Bredno, S. Grzybowsky, and H-G. Machens, *Bildverarbeitung für die medizin 2005*, ch. Vessel Segmentation for Angiographic Enhancement and Analysis, pp. 173–177, Springer Berlin Heidelberg, 2005.

[24] A. P. Condurache, T. Aach, K. Eck, J. Bredno, and T. Stehle, *Fast and robust diaphragm detection and tracking in cardiac X-ray projection images*, SPIE Medical Imaging 2005: Image Processing, 2005, pp. 1766–1775.

[25] P. Condurache, T. Aach, A. Kaiser, and P. Radke, *User-defined ROI tracking for estimation of the myocardial blush grade*, Southwest Symposium on Image Analysis and Interpretation 2006, 2006, pp. 66–70.

[26] C.H. Coombs, R.M. Dawes, and A. Tversky, *Mathematical psychology: An elementary introduction*, Prentice-Hall, New Jersey, 1970.

[27] C. Davatzikos, J.L. Prince, and R.N. Bryan, *Image registration based on boundary mapping*, IEEE Transactions on Medical Imaging **15** (1996), no. 1, 112–115.

[28] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational geometry: Algorithms and applications*, Springer-Verlag, Heidelberg, 1997.

[29] R. Deriche, *Recursively implementing the gaussian and its derivatives*, Tech. report, INRIA, 1993.

[30] J. Duchon, *Interpolation des fonctions de deux variables suivant le principe de la flexion des plaques minces*, RAIRO Analyse Numérique **10** (1976), 5–12.

[31] A.M. Elgammal, D. Harwood, and L.S. Davis, *Non-parametric model for background subtraction*, ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II, Springer-Verlag, 2000, pp. 751–767.

[32] J. Feldmar and N. Ayache, *Rigid and affine registration of smooth surfaces using differential properties*, ECCV (2), 1994, pp. 397–406.

[33] J. Feldmar and N. Ayache, *Rigid, affine and locally affine registration of free-form surfaces*, International Journal of Computer Vision **18** (1996), no. 2, 99–119.

[34] A. Fitzgibbon, *Robust registration of 2D and 3D point sets*, Proceedings of the British Machine Vision Conference, vol. 2, 2001, pp. 411–420.

[35] A.F. Frangi, W.J. Niessen, K.L. Vincken, and M.A. Viergever, *Multiscale vessel enhancement filtering*, Lecture Notes in Computer Science **1496** (1998), 130–138.

[36] W. T. Freeman and E. H. Adelson, *The design and use of steerable filters*, IEEE Transactions on Pattern Analysis and Machine Intelligence **13** (1991), no. 9, 891–906.

[37] H. Fujita., K. Doi, and M. E. Giger, *Investigation of basic imaging properties in digital radiography: 6. MTF's of 11-TV, digital imaging systems*, Medical Physics **12** (1985), 713–720.

[38] C.M. Gibson, C.P. Cannon, S.A. Murphy, K.A. Ryan, R. Mesley, S.J. Marble, C.H. McCabe, F. Van de Werf, and E. Braunwald, *Relationship of TIMI myocardial perfusion grade to mortality after administration of thrombolytic drugs*, Circulation **101** (2000), 125–130.

[39] H. Gray, *Anatomy of the human body*, Lead & Febiger/Bartleby.com, Philadelphia, 1918/2000, http://www.bartleby.com/107/.

[40] J. Gu, C. Toumoulin, and H. Shu, *Spatio-temporal registration in coronary angiography*, Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE, vol. 1, 2003, pp. 584–587.

[41] D. Guo, *Intravascular ultrasound speckle statistics*, Proceedings of the 20th Annual Conference of the IEEE Engineering in Medicine and Biology Society, IEEE, 1998, pp. 796–799.

[42] D. Guo and P. Richardson, *Automatic vessel extraction from angiogram images*, IEEE Computers in Cardiology **25** (1998), 441–4.

[43] J.V. Hajnal, D.L. Hill, and D.J. Hawkes, *Medical image registration*, CRC Press, 2001.

[44] K. Haris, S. Efstratiadis, N. Maglaveras, J. Gourassas, and G. Louridas, *Artery skeleton extraction using topographic and connected component labeling*, Proceedings of the 2001 International Conference on Image Processing, vol. 2, 2001, pp. 339–342.

[45] K. Haris, S. N. Efstratiadis, N. Maglaveras, C. Pappas, J. Gourassas, and G. Louridas, *Model-based morphological segmentation and labeling of coronary angiograms.*, IEEE Trans Med Imaging **18** (1999), no. 10, 1003–15.

[46] K. Haris, N. Maglaveras, S.N. Efstratiadis, J. Gourassas, C. Pappas, and G. Louridas, *Artery skeleton extraction based on consistent curvature labeling*, Computers in Cardiology 1999, 1999, pp. 269–272.

[47] American heart association, *Heart and stroke facts: The heart and how it works*, 2003.

[48] M. Hernández-Hoyos, M. Orkisz, P. Puech, C. Mansard-Desbleds, P Douek, and I.E. Magnin, *Computer-assisted analysis of three-dimensional MR angiograms.*, Radiographics **22** (2002), no. 2, 421–36.

[49] L. Ibáñez, W. Schroeder, L. Ng, and J. Cates, *The ITK software guide*, The Insight Software Consortium, 2005, Available at `http://www.itk.org/ItkSoftwareGuide.pdf`.

[50] M. Kass, A. Witkin, and D. Terzopoulos, *Snakes: Active contour models*, International Journal of Computer Vision **1** (1988), no. 4, 321–331.

[51] E. Kerrien, M-O. Berger, É. Maurincomme, L. Launay, R. Vaillant, and L. Picard, *Fully automatic 3D/2D subtracted angiography registration*, International Conference for Medical Image Computing and Computer Assisted Intervention - MICCAI'99 volume 1679 of Lecture Notes in Computer Science, Springer-Verlag, 1999, pp. 664–671.

[52] R. Keys, *Cubic convolution interpolation for digital image processing*, IEEE Transactions on Signal Processing, Acoustics, Speech, and Signal Processing **29** (1981), no. 6, 1153–1160.

[53] C. Kirbas and F. Quek, *A review of vessel extraction techniques and algorithms*, ACM Computing Survey **36** (2004), no. 2, 81–121.

[54] D.P. Kottke and Y. Sun, *Segmentation of coronary arteriograms by iterative ternary classification*, IEEE Transactions on Biomedical Engineering **37** (1990), no. 8, 778–785.

[55] J. Kybic, *Elastic image registration using parametric deformation models*, Ph.D. thesis, Swiss Fed. Inst. Technol. Lausanne (EPFL), Switzerland, Lausanne, 2001.

[56] L. Lecornu, C. Roux, and J.J. Jacq, *Extraction of vessel contours in angiograms by simultaneous tracking of the two edges*, IEEE Conference on Engineering in Medicine and Biology, vol. 1, 1994, pp. 678–679.

[57] T.M. Lehmann, C. Gonner, and K. Spitzer, *Survey: interpolation methods in medical image processing*, IEEE Transactions on Medical Imaging **18** (1999), no. 11, 1049–1075.

[58] B. Likar and F. Pernuš, *A hierarchical approach to elastic registration based on mutual information*, Image Vision and Computing **19** (2001), 33–44.

[59] T. Lindeberg, *Edge detection and ridge detection with automatic scale selection*, International Journal of Computer Vision **30** (1998), no. 2, 117–154.

[60] K. J. Liu, J. M. Rubin, M. J. Potel, A. Aisen, S. A. MacKay, R. E. Sayre, and C. E. Anagnostopoulos, *Left ventricular wall motion: its dynamic transmural characteristics*, Surgical Research **36** (1984), no. 1, 25–34.

[61] J.H. Long, J.L. Coatrieux, and R. Collorec, *Motion estimation in digital subtraction angiography*, Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First Century., Proceedings of the Annual International Conference of the IEEE, vol. 2, 1989, pp. 567–568.

[62] J. Maintz and M. Viergever, *A survey of medical image registration*, Medical Image Analysis **2** (1998), no. 1, 1–36.

[63] R. Manniesing, M.A. Viergever, and W.J. Niessen, *Vessel enhancing diffusion: A scale space representation of vessel structures.*, Med Image Anal **10** (2006), no. 6, 815–825.

[64] T. McInerney and D. Terzopoulos, *T-snakes: Topology adaptive snakes*, Medical Image Analysis **4** (2000), no. 2, 73–91.

[65] E.H.W. Meijering, W.J. Niessen, and M.A. Viergever, *Retrospective motion correction in digital subtraction angiography: A review*, IEEE Transactions on Medical Imaging **18** (1999), no. 18, 2–21.

[66] J.C. Messenger, S.Y.J. Chen, J.D. Carroll, J.E.B Burchenal, K. Kioussopoulos, and B.M. Groves, *3D coronary reconstruction from routine single-plane coronary*

angiograms: Clinical validation and quantitative analysis of the right coronary artery in 100 patients, The International Journal of Cardiac Imaging **16** (2000), no. 6, 413–427.

[67] F.A. Mettler, *Essentials of radiology, 2nd edition*, Elsevier Saunders, Philadelphia, PA, 2005.

[68] E.G. Miller, N.E. Matsakis, and P.A. Viola, *Learning from one example through shared densities on transforms*, Computer Vision and Pattern Recognition **1** (2000), 464–471.

[69] C. Molina, G. P. Prause, P. Radeva, and M. Sonka, *3-D catheter path reconstruction from biplane angiography using 3D snakes*, SPIE - Medical Imaging, 1998.

[70] D. Nain, A. Yezzi, and G. Turk, *Vessel segmentation using a shape driven flow*, Proceedings of MICCAI, MICCAI, 2004.

[71] J. O'Brien and N. Ezquerra, *Automated segmentation of coronary vessels in angiographic image sequences utilizing temporal, spatial and structural constraints*, Proceedings of the Third Conference on Visualization in Biomedical Computing., SPIE, 1994.

[72] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Applied Mathematical Sciences, vol. 153, Springer, 2003.

[73] S. Osher and N. Paragios (eds.), *Geometric level set methods in imaging*, Vision and Graphics, Springer-Verlag, 2002.

[74] N. Otsu, *A threshold selection method from gray-level histograms*, IEEE Transactions on Systems, Man and Cybernetics **9** (1979), 62–66.

[75] T. Pajdla and L. van Gool, *Matching of 3-D curves using semi-differential invariants*, pp. 390–395, IEEE, 1995.

[76] X. Papademetris and J. S. Duncan, *Cardiac image analysis: Motion and deformation*, SPIE Handbook on Medical Imaging - Volume III: Medical Image Processing and Analysis (J. M. Fitzpatrick and M. Sonka, eds.), SPIE, 2000.

[77] S. Periaswamy, *General-purpose medical image registration*, Ph.D. thesis, Department of Computer Science, Dartmouth College, 2003.

[78] S. Periaswamy and H. Farid, *Elastic registration in the presence of intensity variations*, IEEE Transactions on Medical Imaging **22** (2003), no. 7, 865–874.

[79] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion*, Transactions on Pattern Analysis and Machine Intelligence **12** (1990), no. 7, 629–639.

[80] R. Poli and G. Valli, *An algorithm for real-time vessel enhancement and detection*, Computer Methods and Programs in Biomedicine **52** (1997), 1–22.

[81] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical recipes in C*, 2nd ed., Cambridge University Press, Cambridge, UK, 1992.

[82] Computational Science Education Project, *Mathematical optimization*, 1995, `http://www.phy.ornl.gov/csep/`.

[83] F.K. Quek and C. Kirbas, *Vessel extraction in medical images by wave-propagation and traceback*, IEEE Transactions on Medical Imaging **20** (2001), no. 2, 117–31.

[84] Frédéric J. P. Richard and Laurent D. Cohen, *A new image registration technique with free boundary constraints: application to mammography*, Computer Vision and Image Understanding **89** (2003), no. 2-3, 166–196.

[85] R. A. Robergs and R. Landwehr, *The surprising history of the "HR_{max}=220-age" equation*, Journal of Exercise Physiology **5** (2002), no. 2, 1–10.

[86] D. Ruan, J. A. Fessler, M. Roberson, J. Balter, and M. Kessler, *Nonrigid registration using regularization that accommodates local tissue rigidity*, Proceedings of SPIE: Medical Imaging, 2006.

[87] S. Ruan, A. Bruno, and J-L. Coatrieux, *Three-dimensional motion and reconstruction of coronary arteries from biplane cineangiography*, Image Vision Computing **12** (1994), no. 10, 683–689.

[88] D.B. Rubin and A.P.Dempster, *Maximum likelihood from incomplete date via the EM algorithm*, Journal of the Royal Statistical Society **99** (1977), no. 1, 1–38.

[89] T. J. Ryan, *The coronary angiogram and its seminal contributions to cardiovascular medicine over five decades*, Circulation **106** (2002), 752–756.

[90] I.J. Schoenberg, *Contributions to the problem of approximation of equidistant data by analytic functions*, Quart. Appl. Math. **4** (1946), 45–99 and 112–141.

[91] C. R. Shelton, *Morphable surface models*, International Journal of Computer Vision **38** (2000), no. 1, 75–91.

[92] K.K. Shung, M.B. Smith, and B.M.W. Tsui, *Principles of medical imaging*, Academic Press, 1992.

[93] R. Sinkhorn, *A relationship between arbitrary positive matrices and doubly stochastic matrices*, Ann. Math. Statist. **35** (1964), 876–879.

[94] M. Sonka, G. K. Reddy, M. D. Winniford, and S. M. Collins, *Adaptive approach to accurate analysis of small-diameter vessels in cineangiograms*, IEEE Transactions on Medical Imaging **16** (1997), no. 1, 87–95.

[95] C.O.S. Sorzano, P. Thévenaz, and M. Unser, *Elastic registration of biological images using vector-spline regularization*, IEEE Transactions on Biomedical Engineering **52** (2005), 652–663.

[96] S.A. Stansfiled, *Angy: A rule-based expert system for automatic segmentation of coronary vessels from digital subtracted angiograms*, PAMI **8** (1986), 188–199.

[97] I. Stuke, S. Mollus, A. Condurache, K. Eck, and T. Aach, *Cardio dynamic subrtaction angiography (CDSA)*, Proceedings of the Second Joint EMBS/BMES Conference, IEEE, 2002, pp. 915–916.

[98] J-P. Thirion, *Nonrigid matching using demons*, Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition (CVPR '96), 1996, p. 245.

[99] J-P. Thirion, *Image matching as a diffusion process: an analogy with Maxwell's demons*, Medical Image Analysis **2** (1998), no. 3, 243–260.

[100] A.N. Tikhonov, *Regularization of incorrectly posed problems*, Soviet Math. Dokl. **4** (1963), no. 6, 1624–1627.

[101] The TIMI Study Group, *Thrombolysis in myocardial infarction (TIMI) trial.*, New England Journal of Medicine **312** (1985), 932–6.

[102] D. Tomazevic, B. Likar, T. Slivnik, and F. Pernuš, *3-D/2-D registration of CT and MR to X-ray images*, IEEE Transactions on Medical Imaging **22** (2003), 1407–1416.

[103] S. Ullman and A. Sha'ashua, *Structural saliency: The detection of globally salient structures using a locally connected network*, Tech. Report AIM-1061, Artifical Intelligence Laboratory, MIT, 1988.

[104] M. Unser, *Splines: A perfect fit for signal and image processing*, IEEE Signal Processing Magazine **16** (1999), no. 6, 22–38.

[105] A.W.J. van 't Hof and A. Liem et al., *Angiographic assessment of myocardial reperfusion in patients treated with primary angioplasty for acute myocardial infarction: Myocardial blush grade.*, Circulation **97** (1998), no. 23, 2302–2306.

[106] B.C. Vemuri, J. Ye, Y. Chen, and C.M. Leonard, *Image registration via level-set motion: applications to atlas-based segmentation.*, Medical Image Analysis **7** (2003), no. 1, 1–20.

[107] L. Vincent, *Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms*, IEEE Transactions on Image Processing **2** (1993), 176–201.

[108] E. R. Ritenour W. R. Hendee, *Medical imaging physics, 4th edition*, Wiley, 2002.

[109] G. Wahba, *Spline models for observational data*, Society for Industrial and Applied Mathematics, Philadelphia, 1990.

[110] S.L. Wood, G. Qu, and L.W. Roloff, *Detection and labeling of retinal vessels for longitidunal studies*, IEEE International Conference on Image Processing, vol. 3, 1995, pp. 164–167.

[111] R.P. Woods, J.C. Mazziotta, and S.R. Cherry, *MRI-PET registration with automated algorithm*, Journal of Computer Assisted Tomography **17** (1993), 536–546.

[112] G. W. Zack, W. E. Rogers, and S. A. Latt, *Automatic measurement of sister chromatid exchange frequency*, J Histochem Cytochem **25** (1977), no. 7, 741–753.

[113] Z. Zhang, *Iterative point matching of free-form curves*, Tech. Report 1658, INRIA, 1994.

[114] Y. Zheng and D. Doermann, *Robust point matching for nonrigid shapes by preserving local neighborhood structures*, Transactions on Pattern Analysis and Machine Intelligence **28** (2006), no. 4, 643–649.

[115] A. P. Zijdenbos, B. M. Dawant, R. A. Margolin, and A. C. Palmer, *Morphometric analysis of white matter lesions in mr images: Method and validation*, IEEE Transactions on Medical Imaging **13** (1994), no. 4, 716–724.

[116] L.M. Zir, S.W. Miller, R.E. Dinsmore, J.P. Gilbert, and J.W. Harthorne, *Interobserver variability in coronary angiography*, Circulation **53** (1976), 627–632.

[117] B. Zitová and J. Flusser, *Image registration methods: a survey*, Image and Vision Computing **21** (2003), no. 11, 977–1000.